DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA

MINISERY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY AMAR TELIDJI – LAGHOUAT (UATL)

UNIVERSITY INCUBATOR - LAGHOUAT

DEPARTMENT of ELECTRONICS

FIELD of TELECOMMUNICATIONS

Master Dissertation in
Systems of Telecommunications

PROJECT MANUSCRIPT TO OBTAIN A MASTER/STARTUP DIPLOMA
UNDER THE MINISTERIAL DECREE 1275

MERGING DISTRIBUTED SYSTEM TECHNOLOGY WITH STRATEGIC BUSINESS PLAN: SUFRAH, A FOOD DELIVERY APPLICATION

Students

AKROUT Mohammed Lamine, ATTATI Moussa, SANDOUG Abdlazziz

Jury

REGUIGUE Mourad	MCA	PRESIDENT	UATL
MERAD MESRI Mokhtaria	PROF	SUPERVISER	UATL
BIRANE Mouhoub	MCA	EXAMINER	UATL
LABRAGUE Riyadh Mohammed	MCB	EXAMINER	UATL

Academic Year 2023/2024

Acknowledgments

At the culmination of this humble endeavor, we extend our profound gratitude to Allah, the Almighty, for bestowing upon us the courage, strength, and perseverance needed to complete this work. Without His divine guidance and blessings, this achievement would not have been possible.

We are deeply indebted to our esteemed supervisor, **MESRI MERAD Mokhtaria**, for her unwavering support, insightful guidance, and invaluable advice throughout this journey. Her expertise and dedication have been instrumental in shaping the direction and quality of our work. We also appreciate the interesting and relevant documents she provided, which enriched our research and broadened our understanding.

Our heartfelt thanks also go to the Members of the jury who graciously agreed to review and evaluate our work. We are grateful for their time, effort, and critical insights, which have contributed significantly to the refinement and enhancement of our dissertation.

Additionally, we would like to express our appreciation to our families and friends for their constant support, encouragement, and understanding. Their patience and belief in us have been a source of strength and motivation.

Finally, we acknowledge all those who, directly or indirectly, have supported us in this academic pursuit. Their contributions, no matter how small, have been crucial in helping us reach this milestone.

Thank you all for your invaluable support and guidance.

Dedications

We dedicate this work to our families and friends, for their unwavering support and encouragement, we extend great thanks and appreciation.

Dear father, mother, beloved brothers, and sisters, your steadfast support and belief in our noble pursuit of knowledge and greatness have been invaluable since day one. From the moment our parents instilled in us the importance of seeking wisdom and striving for greatness, we understood that understanding the workings of this world and addressing these fundamental questions is a profound motivation. As a wise person once said, life revolves around simple concepts but requires a complex approach, driven by profound questions that lead to significant quests.

To the entire project team, we deeply appreciate your dedication, hard work, and exceptional teamwork throughout this endeavor. Your commitment and collaborative spirit have been integral to our success. Together, we've overcome challenges, explored new ideas, and achieved milestones that exceed expectations. Your efforts have propelled this project forward and fostered an innovative environment where creativity thrives. Thank you for your outstanding commitment and camaraderie. Here's to celebrating our achievements and inspiring each other in future endeavors.

To the university Ammar Telidji and university Incubator of Laghouat, for their unwavering commitment in providing the resources and support necessary for this project. their dedication has empowered us to pursue our goals of knowledge and greatness, and we are grateful for the opportunities they have afforded us.

Team SUFRAH.

ملخص

نستكشف في هذه المذكرة دراسة مشروع **"سُفْرَةْ"** (SUFRAH) من الجانب العملي لها والذي هو عبارة عن خطة أعمال (Business Plan) و تصميم نموذج أولي لهذا التطبيق الإلكتروني, هذا الأخير مختص في توصيل الطعام الجاهز لفائدة زبائنه.

نتطرق في هذه المذكرة على أهم عناصر الانظمة الموزعة (Distributed System) و الشبكات السلكية و اللاسلكية من بينهم هندسة العميل-الخادم (Client-Server Architecture) لأنه عنصر مهم في الحوسبة الموزعة (Distributed Computing).

نتناول في هذه المذكرة بعض القضايا الحرجة مثل التمركز(Centralization) ، وقابلية التوسع(Scalability) ، والأداء(Performance) ، وتعقيد الاتصال (Communication Complexity) ، مقدمة حلولًا تعزز الأداء العام للنظام ومرونته.

بالإضافة إلى ذلك، توضح هذه المذكرة دراسة الجانب الإستراتيجي الضروري لتطوير واستدامة (SUFRAH) **"سُفْرَةْ"** بنجاح، بما في ذلك تطوير خطة عمل متكاملة. نجد في هذه الدراسة التطبيقية، منهجية مُنظمة لتحديد الإيرادات الممكنة واستراتيجيات التسويق و خطط التشغيل، مما يضمن النجاح على المدى الطويل للشركة الناشئة.

من خلال دمج مكتسباتنا في مجال الاتصالات مع الهياكل الحاسوبية الحديثة، تظهر هذه المذكرة كيف يمكن لـ (SUFRAH) **"سُفْرَةْ"** الاستفادة بشكل ماهر من التقنيات الحديثة لتوفير خدمات توصيل الطعام بشكل سلس وقابل للتوسع و التنوع. يضع النهج الشامل والبيني لتطبيق **"سُفْرَةْ"** (SUFRAH) في موقع متميز لتلبية متطلبات سوق ناشئة، وتقديم حلول فعالة وآمنة وذات أداء عالي في ميدان توصيل الطعام.

كلمات مفتاحية

أداء نظام، أمن بيانات، استدامة، تعقيد تواصل، تمركز، حلول قابلة توسع، حوسبة موزعة، ميدان توصيل طعام، هندسة عميل وخادم، هياكل حوسبة.

Abstract

This dissertation explores the development of SUFRAH, an advanced food delivery application that integrates sophisticated technical architectures with strong business strategies.

The work investigates through distributed system and networking, The study addresses critical issues such as centralization, scalability, performance, and communication complexity, proposing solutions that enhance overall system performance and flexibility.

Additionally, the study outlines the strategic framework necessary for the successful implementation and sustainability of SUFRAH, including the development of a business plan.

This approach provides a structured methodology for identifying viable revenue models, marketing strategies, and operational plans, ensuring the application's long-term success.

By integrating telecommunications expertise with cutting-edge computing architectures, this work demonstrates how SUFRAH can skillfully leverage modern technologies to provide seamless and scalable food delivery services. The comprehensive and interdisciplinary approach positions SUFRAH to meet the demands of a burgeoning market, offering efficient, secure, and high-performance solutions within the food delivery industry.

KEY WORDS

Business strategies, Centralization, Client-server architecture, Communication Complexity, Computing Architectures, Distributed Computing, Food delivery industry, Scalability, Strategic Framework, Sustainability, System Performance, Technical architectures.

Table of contents

List of Figures

List of Tables

List of abbreviations

General introduction

Introduction

Computer networks are ubiquitous, encompassing the Internet and its many constituent networks, as well as mobile phone networks, corporate networks, factory networks, campus networks, home networks, and in-car networks. Each of these, individually and collectively, exhibit the core characteristics that qualify them as distributed systems.

Distributed systems enable resource sharing, fault tolerance, and scalability, allowing for efficient and resilient operations across various applications. This desertion We will explore topics such as failure models, providing an overview in distributed systems.

Problem Statement

1. Traditional IT (Information Technology) systems have long faced significant challenges due to their centralized nature, leading to substantial security and data management issues.

2. These systems are particularly vulnerable to single points of failure, which can cause widespread disruptions.

3. Furthermore, monolithic systems struggle to scale and maintain performance as user demands increase, and they often encounter logistical challenges in deployment and maintenance.

4. Additionally, the complexity involved in communication between distributed system components hinders overall efficiency, while ensuring quality assurance becomes increasingly difficult.

5. Moreover, traditional systems provide limited options for customization and flexibility, constraining organizations in adapting to evolving technological landscapes.

These compounded issues underscore the critical need for reevaluating traditional IT architectures to address security, scalability, communication inefficiencies, logistical challenges, quality assurance, and flexibility.

Objectives of the Study

Our dissertation serves the following goals:

1. Comprehensive Analysis of Distributed Systems: In-depth study of distributed systems with special emphasis on client-server architecture.

2. Detailed Examination of Networking and Client-Server Architecture: Analyze the client-server model by detailing its hierarchical structure, functional components, and communication protocols.

3. Case Study on Google Search Engine: Use the Google search engine as a case study to illustrate the theoretical foundations, architectural principles and practical applications of distributed systems.

4. Development of a Business Plan: develop a comprehensive business plan for the SUFRAH which is a startup in food delivery field , This will include identifying target market segments, assessing competitive intensity, devising strategic plans for product positioning, promotion, and sales tactics, detailing production processes and workforce planning, establishing key partnerships, conducting thorough financial analysis, managing cash flow meticulously, ensuring operational efficiency through optimal resource allocation, and performing rigorous risk assessment to mitigate potential threats.

Methodology Overview

This study adopts an innovative approach by leveraging the latest advancements in distributed systems technology, drawing inspiration from industry leaders such as Google. The methodology is structured into two primary phases: The development of the business plan and the theoretical aspect of networking in distributed system.

Relevance and Application

The relevance of these architectures extends to wireless communications and mobile phones, where they support the development of applications capable of efficient communication over wireless networks. In particular, this study focuses on the application of these technologies to the SUFRAH food delivery application. By distributing data, improving performance, enabling scalability, and streamlining communication, SUFRAH can provide efficient, secure, and high-performance food delivery services.

Structure of the Dissertation

This dissertation is structured as follows:

Chapter 1: chapter 1: This chapter investigates the client-server architecture by detailing its roles, benefits, design principles, and common applications in distributed computing. It delves into essential protocols and infrastructure, explaining the OSI (Open System Interconnection) model and the importance of switching technologies such as Ethernet and packet switching for efficient data transfer. The chapter also emphasizes the significance of TCP/IP TCP/IP (Transmission Control Protocol/Internet Protocol) in ensuring reliable global network communication. Furthermore, it examines distributed systems through a case study of the Google search engine and discusses failure models, offering a comprehensive foundation for understanding distributed systems and their practical applications.

*Chapter 2***:** This chapter We will explore the various aspects of our project in detail, starting with an introduction to our project idea and the values that underpin it. We will then describe our team, our objectives, and our implementation schedule, highlighting the innovative aspects of our approach. Our structured document thoroughly outlines the identification of target market segments and the assessment of competitive intensity, strategic plans for service positioning, promotion strategies, and engagement tactics. It includes detailed descriptions of production processes, comprehensive workforce planning, key partnerships with crucial external collaborators and suppliers, thorough financial analysis including cost breakdowns, revenue projections, and expected financial outcomes, meticulous cash flowmanagement strategies, a focus on operational efficiency through optimal resource allocation and well-defined organizational structures, and thorough risk assessment to identify potential threats and mitigation strategies. All these elements are intricately aligned with overarching strategic business objectives for clarity and coherence. Our goal is clear: to offer a viable and sustainable service that will contribute to the economic and social development of our community.

CHAPTER I

Foundations of Networking and Distributed Systems

CHAPTER I. Foundations of Networking and Distributed Systems

I.1 Introduction

Distributed computing has become an essential paradigm in the development and implementation of modern information systems, enabling the distribution of tasks and resources across multiple computers. At the center of this paradigm is the client-server architecture, a fundamental technology that enables efficient and effective communication, data management, and resource allocation in distributed systems.

Client-server architecture was established in 1994 to create a communication framework for centralizing information securely. It operates with clients, typically end-user applications, requesting services and resources from centralized servers. This architecture addresses critical issues encountered with previous centralized systems, such as security, data management, and scalability. By centralizing services and data, the client-server model improves control, enhances data security, and simplifies the maintenance of complex systems.

Client-server architecture is an established model in distributed computing where clients, usually end-user applications, request services and resources from centralized servers. This architecture solves several critical problems encountered with previous centralized systems, such as security, data management and scalability. By distribute services and data, the client-server model improves control, increases data security and simplifies the maintenance of complex systems.

The processing speed of workstations in such architecture is increased thanks to the server structure that contains several CPUs (Central Processing Units). Capacity and memory are increased. It is safe to say that client-server technology has grown exponentially in all areas of activity, including:

- Database management.
- Transactional systems.
- Email, web, Internet systems.
- Data sharing systems.

In the context of distributed computing, understanding the types of networks, network principles, and IP is crucial. Different types of networks (such as LANs (Local Area Network), WANs (Wide Area Network), and MANs (Metropolitan Area Network)) and network principles (like scalability, reliability, and latency) play integral roles in designing and deploying distributed systems. Internet Protocol forms the backbone of network communication, facilitating the transmission of data packets across interconnected networks.

This chapter explores the client-server architecture, focusing on its roles, benefits, and applications in distributed computing. It begins with an overview of the client-server model, detailing its design principles, advantages, and common use cases.

Additionally, the chapter delves into the protocols and infrastructure essential for distributed computing. It explains the OSI model and highlights the importance of switching technologies like Ethernet and packet switching for efficient data transfer. The significance of TCP/IP in ensuring reliable global network communication is also covered. Furthermore, it includes an examination of distributed systems, illustrated through a case study of the Google search engine and a discussion on failure models. This comprehensive overview provides a solid foundation for understanding distributed systems and their practical applications.

I.2 Type of networks

Here we introduce the main types of networks that are used to support distributed systems: personal area networks, local area networks, wide area networks, metropolitan area networks and the wireless variants of them. Internetworks such as the Internet are constructed from networks of all these types. *Table I.1* shows the performance characteristics of the various types of networks discussed below. *Table I.1*

Some of the names used to refer to types of networks are confusing because they seem to refer to the physical extent (local area, wide area), but they also identify physical transmission technologies and low-level protocols. These are different for local and wide area networks, although some network technologies, such as ATM (Asynchronous Transfer Mode), are suitable for both local and wide area applications and some wireless networks also support local and metropolitan area transmission.

We refer to networks that are composed of many interconnected networks, integrated to provide a single data communication medium, as internetworks. The Internet is the prototypical internetwork; it is composed of millions of local, metropolitan and wide area networks. (Coulouris, G.Dollimore, Kindberg, & G., 2012)

Table I-1 Network performance

	Example	**Range**	**Bandwidth (Mbps)**	**Latency (ms)**
Wired:				
LAN	Ethernet	1–2 kms	10–10,000	1–10
WAN	ATM	worldwide	0.010–600	100–500
MAN	IP routing	2–50 kms	1–600	10
Internetwork	Internet	worldwide	0.5–600	100–500
Wireless:				
WPAN	Bluetooth (IEEE 802.15.1)	10–30m	0.5–2	5–20
WLAN	WiFi (IEEE 802.11)	0.15–1.5 km	11–108	5–20
WMAN	WiMAX (IEEE 802.16)	5–50 km	1.5–20	5–20
WWAN	5G phone	cell: 0.5 km	70 – 205	5–36

I.3 Network principles

The basis for all computer networks is the packet-switching technique first developed in the 1960s. This enables data packets addressed to different destinations to share a single communications link, unlike the circuit-switching technology that underlies conventional telephony. Packets are queued in a buffer and transmitted when the link is available. Communication is asynchronous – messages arrive at their destination after a delay that varies depending upon the time that packets take to travel through the network. (Coulouris, G.Dollimore, Kindberg, & G., 2012)

a. Packet transmission

In most applications of computer networks the requirement is for the transmission of logical units of information, or messages – sequences of data items of arbitrary length. But before a message is transmitted it is subdivided into packets. The simplest form of packet is a sequence of binary data (an array of bits or bytes) of restricted length, together with addressing information sufficient to identify the source and destination computers. Packets of restricted length are used:

• so that each computer in the network can allocate sufficient buffer storage to hold the largest possible incoming packet;

• to avoid the undue delays that would occur in waiting for communication channels to become free if long messages were transmitted without subdivision.

b. Data streaming

The transmission and display of audio and video in real time is referred to as streaming. It requires much higher bandwidths than most other forms of communication in distributed systems. Multimedia applications rely upon the transmission of streams of audio and video data elements at guaranteed high rates and with bounded latencies.

A video stream requires a bandwidth of about 1.5 Mbps if the data is compressed, or 120 Mbps if uncompressed. UDP (User Datagram Protocol) internet packets are generally used to hold the video frames, but because the flow is continuous as opposed to the intermittent traffic generated by typical client-server interactions, the packets are handled somewhat differently. The play time of a multimedia element such as a video frame is the time at which it must be displayed (for a video element) or converted to sound (for a sound sample). For example, in a stream of video frames with a frame rate of 24 frames per second, frame N has a play time that is N/24 seconds after the stream's start time. Elements that arrive at their destination later than their play time are no longer useful and will be dropped by the receiving process.

c. Switching schemes

A network consists of a set of nodes connected together by circuits. To transmit information between two arbitrary nodes, a switching system is required. Here we define the four types of switching that are used in computer networking.

Broadcast: Broadcasting is a transmission technique that involves no switching. Everything is transmitted to every node, and it is up to potential receivers to notice transmissions addressed to them. Some LAN technologies, including Ethernet, are based on broadcasting. Wireless networking is necessarily based on broadcasting, but in the absence of fixed circuits the broadcasts are arranged to reach nodes grouped in cells.

Circuit switching: At one time telephone networks were the only telecommunication networks. Their operation was simple to understand: when a caller dialed a number, the pair of wires from her phone to the local exchange was connected by an automatic switch at the exchange to the pair of wires connected to the other party's phone. For a long-distance call the process was similar but the connection would be switched through a number of intervening exchanges to its destination. This system is sometimes referred to as the plain old telephone system. It is a typical circuit-switching network.

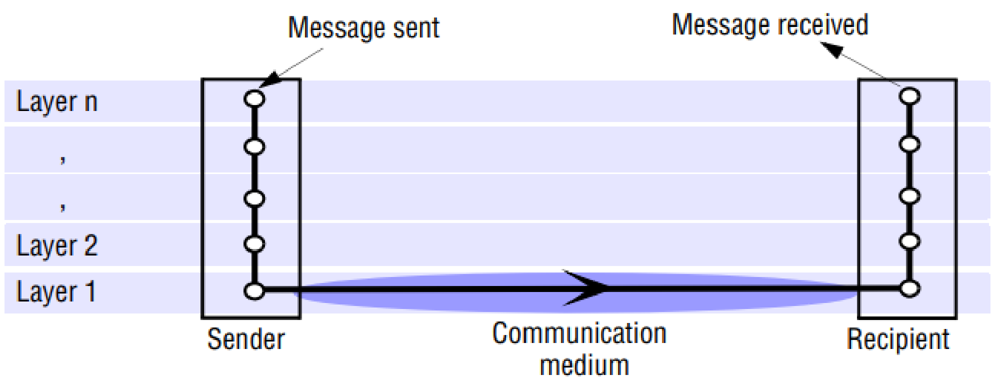


Figure I.1 Conceptual layering of protocol software

Packet switching: The advent of computers and digital technology brought many new possibilities for telecommunication. At the most basic level, it brought processing and storage.

These made it possible to construct a different kind of communication network called a store-and-forward network. Instead of making and breaking connections to build circuits, a store-and-forward network just forwards packets from their source to their destination.

Frame relay: In reality, it takes anything from a few tens of microseconds to a few milliseconds to switch a packet through each network node in a store-and-forward network. This switching delay depends on the packet size, hardware speed and quantity of other traffic, but its lower bound is determined by the network bandwidth, since the entire packet must be received before it can be forwarded to another node. Much of the Internet is based on store-and forward switching, and as we have already seen, even short Internet packets typically take up to 200 milliseconds to reach their destinations. where delays of less than 50 milliseconds are needed to sustain high-quality conversation. Delays of this magnitude are too long for real-time applications such as telephony and video conferencing.

The frame relay switching method brings some of the advantages of circuit switching to packet-switching networks. They overcome the delay problems by switching small packets (called frames) on the fly. The switching nodes (which are usually special-purpose parallel digital processors) route frames based on the examination of their first few bits; frames as a whole are not stored at nodes but pass through them as short streams of bits. ATM networks are a prime example; high-speed ATM networks can transmit packets across networks consisting of many nodes in a few tens of microseconds. (Coulouris, G.Dollimore, Kindberg, & G., 2012)

I.3.2 Protocols

The term protocol is used to refer to a well-known set of rules and formats to be used for communication between processes in order to perform a given task. The definition of a protocol has two important parts to it:

- a specification of the sequence of messages that must be exchanged;
- a specification of the format of the data in the messages.

The existence of well-known protocols enables the separate software components of distributed systems to be developed independently and implemented in different programming languages on computers that may have different order codes and data representations.

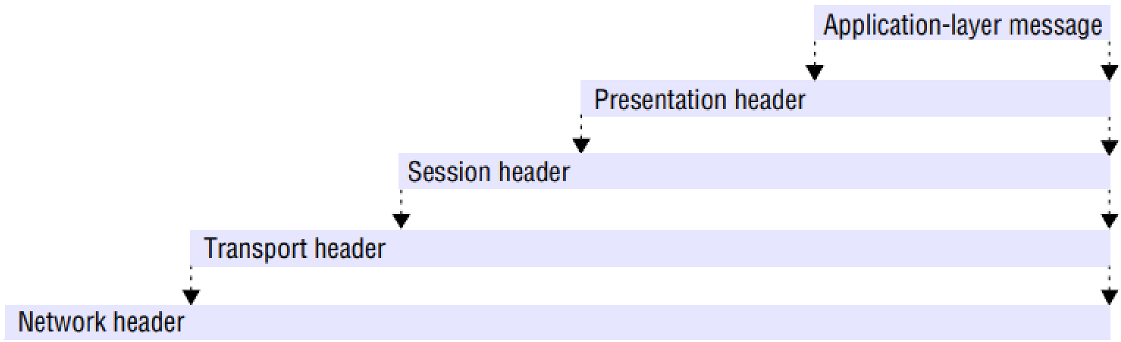


Figure I.2 Encapsulation as it is applied in layered protocols

a. Protocol layers •

Network software is arranged in a hierarchy of layers. Each layer presents an interface to the layers above it that extends the properties of the underlying communication system. A layer is represented by a module in every computer connected to the network. *Figure I.3* illustrates the structure and the flow of data when a message is transmitted using a layered protocol. Each module appears to communicate directly with a module at the same level in another computer in the network, but in reality, data is not transmitted directly between the protocol modules at each level. Instead, each layer of network software communicates by local procedure calls with the layers above and below it.

On the sending side, each layer (except the topmost, or application layer) accepts items of data in a specified format from the layer above it and applies transformations to encapsulate the data in the format specified for that layer before passing it to the layer below for further processing. *Figure I.3*

Application-layer message Presentation header Session header Transport header Network header illustrates this process as it applies to the top four layers of the OSI protocol suite (discussed in the next subsection). The figure shows the packet headers that hold most network-related data items, but for clarity it omits the trailers that are present in some types of packets; it also assumes that the application-layer message to be transmitted is shorter than the underlying network's maximum packet size. If not, it would have to be encapsulated in several network-layer packets. On the receiving side, the converse transformations are applied to data items received from the layer below before they are passed to the layer above. The protocol type of the layer above is included in the header of each layer, to enable the protocol stack at the receiver to select the correct software components to unpack the packets.

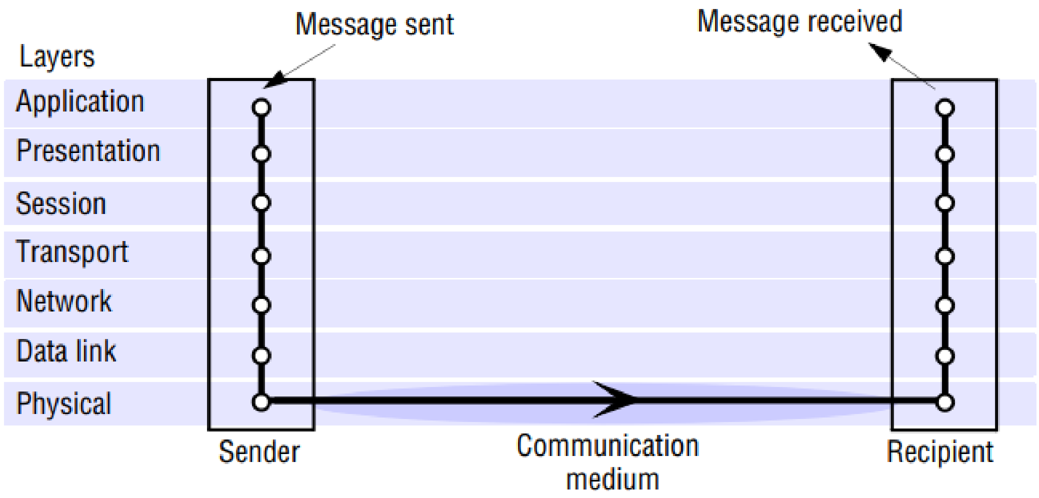


Figure I.3 Protocol layers in the Open Systems Interconnection protocol model

b. Protocol suites

A complete set of protocol layers is referred to as a protocol suite or a protocol stack, reflecting the layered structure. *Figure I.3* shows a protocol stack that conforms to the seven-layer Reference Model for OSI adopted by the ISO (International Organization for Standardization) [ISO 1992]. The OSI Reference Model was adopted in order to encourage the development of protocol standards that would meet the requirements of open systems.

The purpose of each level in the OSI Reference Model is summarized in *Table I.2*. As its name implies, it is a framework for the definition of protocols and not a definition for a specific suite of protocols. Protocol suites that conform to the OSI model must include at least one specific protocol at each of the seven levels that the model defines.

Protocol layering brings substantial benefits in simplifying and generalizing the software interfaces for access to the communication services of networks, but it also carries significant performance costs. The transmission of an application-level message via a protocol stack with N layers typically involves N transfers of control to the relevant layer of software in the protocol suite, at least one of which is an operating system entry, and taking N copies of the data as a part of the encapsulation mechanism. All of these overheads result in data transfer rates between application processes that are much lower than the available network bandwidth.

Table I-2 OSI protocol summary

Layer	Description	Examples
Application	Protocols at this level are designed to meet the communication requirements of specific applications, often defining the interface to a service.	HTTP, FTP, SMTP, CORBA IIOP
Presentation	Protocols at this level transmit data in a network representation that is independent of the representations used in individual computers, which may differ. Encryption is also performed in this layer, if required.	TLS security, CORBA data representation
Session	At this level reliability and adaptation measures are performed, such as detection of failures and automatic recovery.	SIP
Transport	This is the lowest level at which messages (rather than packets) are handled. Messages are addressed to communication ports attached to processes. Protocols in this layer may be connection-oriented or connectionless.	TCP, UDP
Network	Transfers data packets between computers in a specific network. In a WAN or an internetwork this involves the generation of a route passing through routers. In a single LAN no routing is required.	IP, ATM virtual circuits
Data link	Responsible for transmission of packets between nodes that are directly connected by a physical link. In a WAN transmission is between pairs of routers or between routers and hosts. In a LAN it is between any pair of hosts.	Ethernet MAC, ATM cell transfer, PPP
Physical	The circuits and hardware that drive the network. It transmits sequences of binary data by analogue signalling, using amplitude or frequency modulation of electrical signals (on cable circuits), light signals (on fibre optic circuits) or other electromagnetic signals (on radio and microwave circuits).	Ethernet baseband signalling, ISDN

The internetwork layer is a ‘virtual’ network layer that is responsible for transmitting internetwork packets to a destination computer. An internetwork packet is the unit of data transmitted over an internetwork.

Internetwork protocols are overlaid on underlying networks as illustrated in *Figure I.4*. The network interface layer accepts internetwork packets and converts them into packets suitable for transmission by the network layers of each underlying network.

c. Packet assembly

The task of dividing messages into packets before transmission and reassembling them at the receiving computer is usually performed in the transport layer. The network-layer protocol packets consist of a header and a data field. In most network technologies, the data field is variable in length, with the maximum length called the MTU (Maximum Transfer Unit). If the length of a message exceeds the MTU of the underlying network layer, it must be fragmented into chunks of the appropriate size, with sequence numbers for use on reassembly, and transmitted in multiple packets. For example, the MTU for Ethernets is 1500 bytes – no more than that quantity of data can be transmitted in a single Ethernet packet.

Although the IP protocol stands in the position of a network-layer protocol in the Internet suite of protocols, its MTU is unusually large at 64 kbytes (8 kbytes is often used in practice because some nodes are unable to handle such large packets).

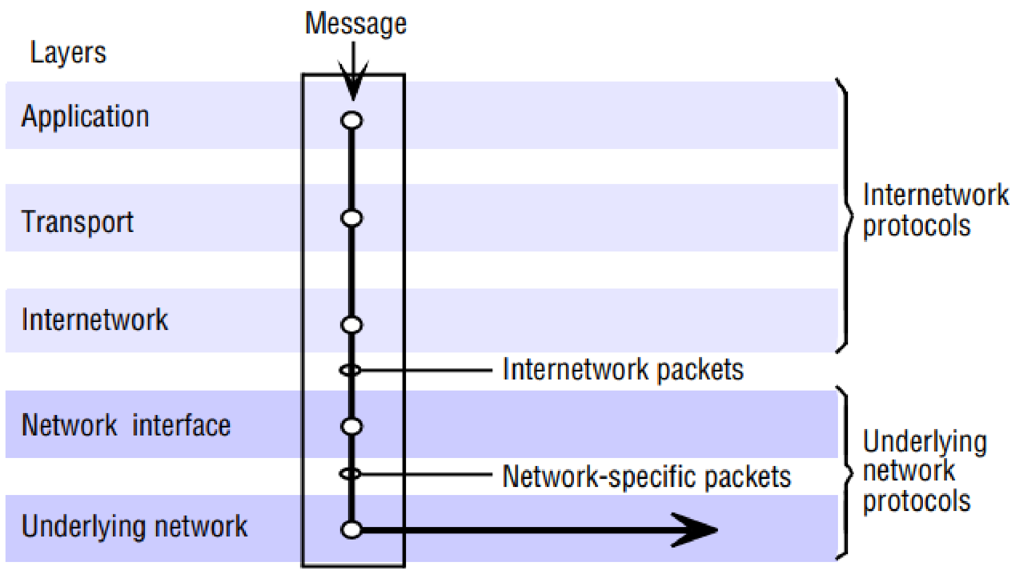


Figure I.4 Internetwork layers

Whichever MTU value is adopted for IP packets, packets larger than the Ethernet MTU can arise and they must be fragmented for transmission over Ethernets.

I.3.3 Internet Protocol

We describe here the main features of the TCP/IP suite of protocols and discuss their advantages and limitations when used in distributed systems. *Figure I.5* TCP/IP layers Messages (UDP) or streams TCP Application Transport Internet UDP or TCP packets IP datagrams Network-specific frames Message Layers Underlying network

Network interface: The Internet emerged from two decades of research and development work on wide area networking in the USA, commencing in the early 1970s with the ARPANET (Advanced Research Projects Agency Network)

The first large-scale computer network development [Leiner et al. 1997]. An important part of that research was the development of the TCP/IP protocol suite. TCP stands for Transmission Control Protocol, IP for Internet Protocol. The widespread adoption of the TCP/IP and Internet application protocols in national research networks, and more recently in commercial networks in many countries, has enabled the national networks to be integrated into a single internetwork that has grown extremely rapidly to its present size, with more than 60 million hosts. Many application services and application-level protocols (shown in parentheses in the following list) now exist based on TCP/IP, including the Web (HTTP (Hypertext Transfer Protocol)), email (SMTP, POP), netnews (NNTP (Network News Transfer Protocol)), file transfer (FTP (File Transfer Protocol)) and Telnet (telnet). TCP is a transport protocol; it can be used to support applications directly, or additional protocols can be layered on it to provide additional features.

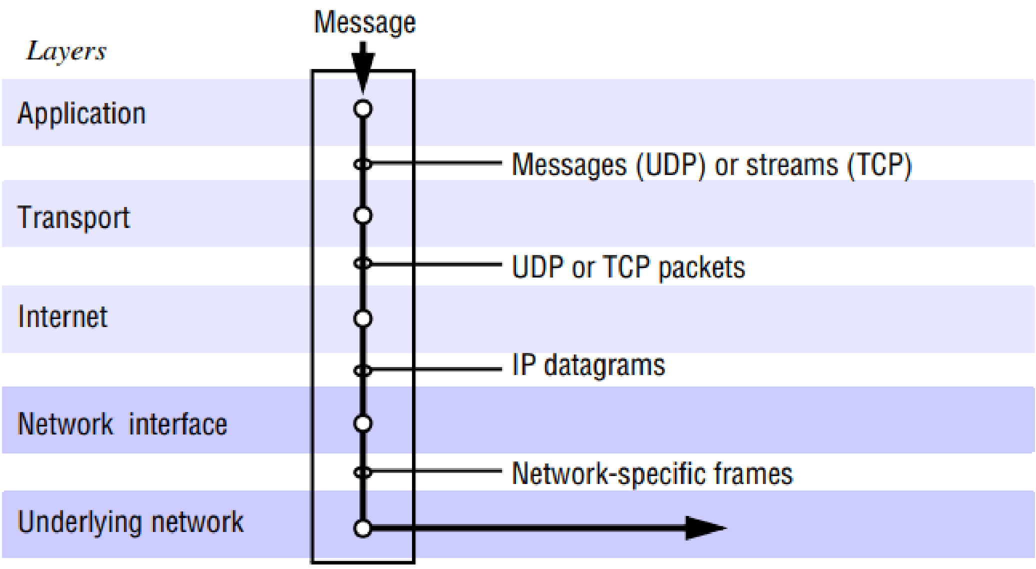


Figure I.5 TCP/IP layers

For example, HTTP is usually transported by the direct use of TCP, but when end-to-end security is required, the TLS (Transport Layer Security) protocol is layered on top of TCP to produce secure channels and HTTP messages are transmitted via the secure channels.

The Internet protocols were originally developed primarily to support simple wide area applications such as file transfer and electronic mail, involving communication with relatively high latencies between geographically dispersed computers, but they turned out to be efficient enough to support the requirements of many distributed applications on both wide area and local networks and they are now almost universally used in distributed systems. The resulting standardization of communication protocols has brought immense benefits.

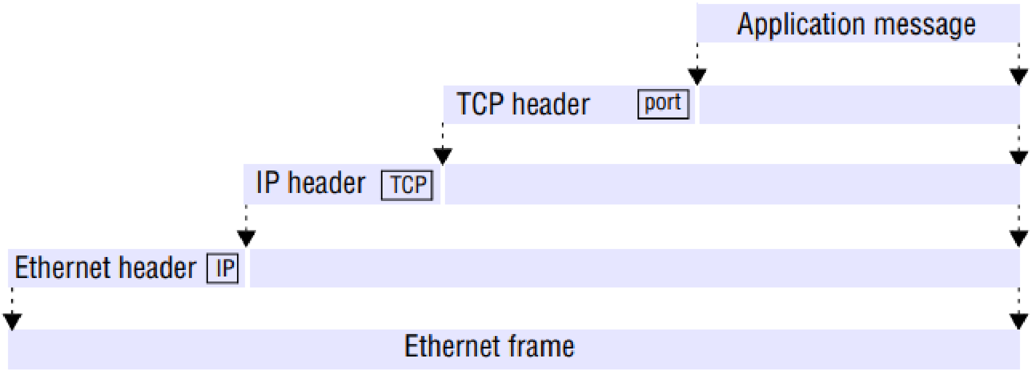


Figure I.6 Encapsulation as it occurs when a message is transmitted via TCP over an Ethernet

The general illustration of internetwork protocol layers of *Figure I.5* is translated into the specific Internet case in *Figure I.5*. There are two transport protocols – TCP and UDP.

TCP is a reliable connection-oriented protocol, and UDP is a datagram protocol that does not guarantee reliable transmission. The Internet Protocol is the underlying 'network' protocol of the Internet virtual network – that is, IP datagrams provide the basic transmission mechanism for the Internet and other TCP/IP networks. We placed the word 'network' in quotation marks in the preceding sentence because it is not the only network layer involved in the implementation of Internet communication. This is because the Internet protocols are usually layered over another network technology, such as Ethernet, which already provides a network layer that enables the computers attached to the same network to exchange datagrams. *Figure I.6* Encapsulation as it occurs when a message is transmitted via TCP over an Ethernet port TCP/IP *Figure I.6* illustrates the encapsulation of packets that would occur for the transmission of a message via TCP over an underlying Ethernet. The tags in the headers are the protocol types for the layers above, needed for the receiving protocol stack to correctly unpack the packets. In the TCP layer, the receiver's port number serves a similar purpose, enabling the TCP software component at the receiving host to pass the message to a specific application-level process.

a. TCP and UDP

TCP and UDP provide the communication capabilities of the Internet in a form that is useful for application programs. Application developers might wish for other types of transport service, for example to provide real-time guarantees or security, but such services would generally require more support in the network layer than IPv4 provides. TCP and UDP can be viewed as a faithful reflection at the application programming level of the communication facilities that IPv4 has to offer. IPv6 is another story; it will certainly continue to support TCP and UDP, but it includes capabilities that cannot be conveniently accessed through TCP and UDP. It may be useful to introduce additional types of transport service to exploit them, once the deployment of IPv6 is sufficiently wide to justify their development.

Use of ports: The first characteristic to note is that, whereas IP supports communication between pairs of computers (identified by their IP addresses), TCP and UDP, as transport protocols, must provide process-to-process communication. This is accomplished by the use of ports. Port numbers are used for addressing messages to processes within a particular computer and are valid only within that computer. A port number is a 16-bit integer. Once an IP packet has been delivered to the destination host, the TCP- or UDP-layer software dispatches it to a process via a specific port at that host.

UDP features: UDP is almost a transport-level replica of IP. A UDP datagram is encapsulated inside an IP packet. It has a short header that includes the source and destination port numbers (the corresponding host addresses are present in the IP header), a length field and a checksum. UDP offers no guarantee of delivery. We have already noted that IP packets may be dropped because of congestion or network error. UDP adds no additional reliability mechanisms except the checksum, which is optional. If the checksum field is non-zero, the receiving host computes a check value from the packet contents and compares it with the received checksum; packets for which they do not match are dropped.

Thus, UDP provides a means of transmitting messages of up to 64 Kbytes in size (the maximum packet size permitted by IP) between pairs of processes (or from one process to several in the case of datagrams addressed to IP multicast addresses), with minimal additional costs or transmission delays above those due to IP transmission. It incurs no setup costs and it requires no administrative acknowledgement messages. But its use is restricted to those applications and services that do not require reliable delivery of single or multiple messages.

TCP features: TCP provides a much more sophisticated transport service. It provides reliable delivery of arbitrarily long sequences of bytes via stream-based programming abstraction. The reliability guarantee entails the delivery to the receiving process of all of the data presented to the TCP software by the sending process, in the same order. TCP is connection-oriented. Before any data is transferred, the sending and receiving processes must cooperate in the establishment of a bidirectional communication channel. The connection is simply an end-to-end agreement to perform reliable data transmission; intermediate nodes such as routers have no knowledge of TCP connections, and the IP packets that transfer the data in a TCP transmission do not necessarily all follow the same route

The TCP layer includes additional mechanisms (implemented over IP) to meet the reliability guarantees. These are:

Sequencing: A TCP sending process divides the stream into a sequence of data segments and transmits them as IP packets. A sequence number is attached to each TCP segment. It gives the byte number within the stream for the first byte of the segment. The receiver uses the sequence numbers to order the received segments before placing them in the input stream at the receiving process. No segment can be placed in the input stream until all lower-numbered segments have been received and placed in the stream, so segments that arrive out of order must be held in a buffer until their predecessors arrive.

Flow control: The sender takes care not to overwhelm the receiver or the intervening nodes. This is achieved by a system of segment acknowledgements. Whenever a receiver successfully receives a segment, it records its sequence number. From time to time the receiver sends an acknowledgement to the sender, giving the sequence number of the highest-numbered segment in its input stream together with a window size. If there is a reverse flow of data, acknowledgements are carried in the normal data segments; otherwise, they travel in acknowledgement segments. The window size field in the acknowledgement segment specifies the quantity of data that the sender is permitted to send before the next acknowledgement

Retransmission: The sender records the sequence numbers of the segments that it sends. When it receives an acknowledgement, it notes that the segments were successfully received, and it may then delete them from its outgoing buffers. If any segment is not acknowledged within a specified timeout, the sender retransmits it.

Buffering: The incoming buffer at the receiver is used to balance the flow between the sender and the receiver. If the receiving process issues receive operations more slowly than the sender issues send operations, the quantity of data in the buffer will grow. Usually, it is extracted from the buffer before it becomes full, but ultimately the buffer may overflow, and when that happens incoming segments are simply dropped without recording their arrival. Their arrival is therefore not acknowledged and the sender is obliged to retransmit them.

Checksum: Each segment carries a checksum covering the header and the data in the segment. If a received segment does not match its checksum, the segment is dropped. (Coulouris, G.Dollimore, Kindberg, & G., 2012)

I.4 Client-Server Architecture

Client-server architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. Client-server architecture is also known as a networking computing model or client-server network because all the requests and services are delivered over a network.

I.4.1 Definition and principles

The Client-server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters called clients. In the client-server architecture, when the client computer sends a request for data to the server through the internet, the server accepts the requested process and delivers the data packets requested back to the client. Clients do not share any of their resources.

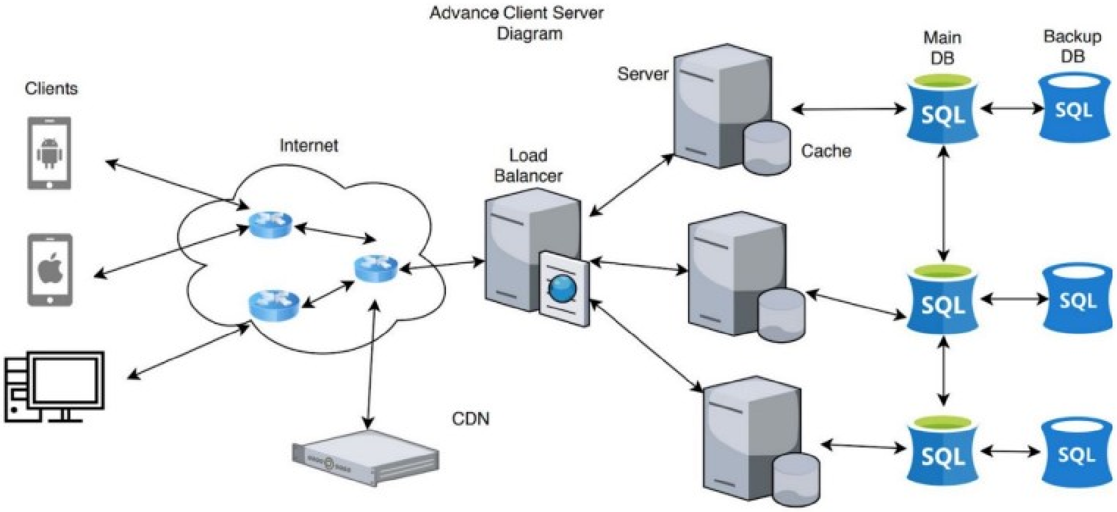


Figure I.7 Client server model (Client-Server Model, 2024)

The flow of the data is unidirectional and forms a cycle. It is usually initiated by the client requesting some kind of data and the server processing the request and sending some kind of data back to the client via a protocol. Clients cannot directly talk to each other. A typical topological data flow goes as follows:

Client requests data from server.

Load balancer routes the request to the appropriate server.

Server processes the request client.

Server queries appropriate database for some data.

Database returns the queried data back to the server.

The server processes the data and sends the data back to the client.

This process repeats. (Client-Server Model, 2024).

I.4.2 Historical context and development

Even though most people use the term "client/server" when talking about group computing with PCs (Personal Computer) on networks, PC network computing evolved before the client/server model started gaining acceptance in the late 1980's. These first PC networks were based on the file sharing metaphor. In file sharing, the server simply downloads or transfers files from the shared location to your desktop where the logic and data for the job run in their entirety. This approach was popularized mostly by Xbase style products (dBASE, FoxPro and Clipper). File sharing is simple and works as long as shared usage is low, update contention is very low, and the volume of data to be transferred is low compared with LAN capacity.

As PC LAN computing moved into the 90's two megatrends provided the impetus for client/server computing. The first was that as first-generation PC LAN applications and their users both grew; the capacity of file sharing was strained. Multi-user Xbase technology can provide satisfactory performance for a few up to maybe a dozen simultaneous users of a shared file, but it's very rare to find a successful implementation of this approach beyond that point.

The second change was the emergence and then dominance of the GUI (Graphical User Interface) metaphor on the desktop. Very soon GUI presentation formats, led by Windows and Mac, became mandatory for presenting information. The requirement for GUI displays meant that traditional mini or mainframe applications with their terminal displays soon looked hopelessly out of date. The architecture and technology that evolved to answer this demand was client/server, in the guise of a two-tiered approach. By replacing the file server with a true database server, the network could respond to client requests with just the answer to a query against a relational DBMS (Data Base Management System) (rather than the entire file). One benefit to this approach, then, is to significantly reduce network traffic. Also, with a real DBMS, true multi-user updating is now easily available to users on the PC LAN. By now, the idea of using Windows or Mac (Macintosh) style PCs to front end a shared database server is familiar and widely implemented. (Schussel, 2001)

I.4.3 Key components (clients, server, network)

The term server is probably familiar. It refers to a running program (a process) on a networked computer that accepts requests from programs running on other computers to perform a service and responds appropriately. The requesting processes are referred to as clients, and the overall approach is known as client-server computing. In this approach, requests are sent in messages from clients to a server and replies are sent in messages from the server to the clients. When the client sends a request for an operation to be carried out, we say that the client invokes an operation upon the server.

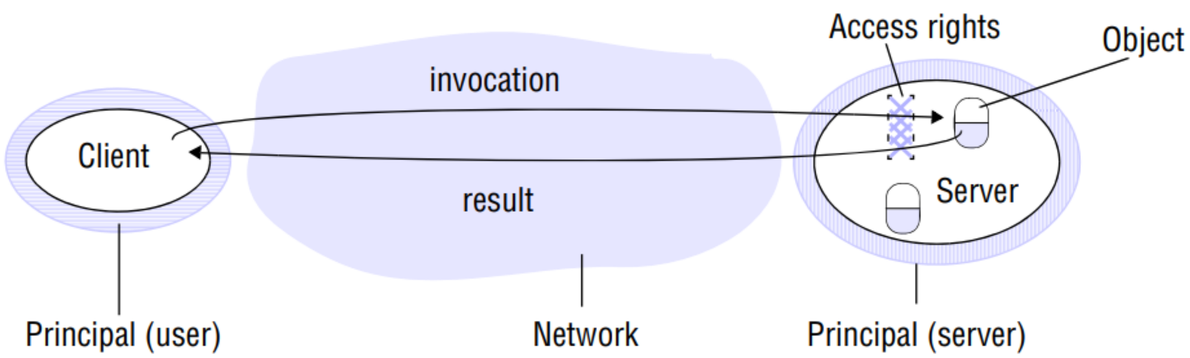


Figure I.8 Principals remote objects invocation

A complete interaction between a client and a server, from the point when the client sends its request to when it receives the server's response, is called a remote invocation. The same process may be both a client and a server, since servers sometimes invoke operations on other servers. The terms 'client' and 'server' apply only to the roles played in a single request. Clients are active (making requests) and servers are passive (only waking up when they receive requests); servers run continuously, whereas clients last only as long as the applications of which they form a part. (Coulouris, G.Dollimore, Kindberg, & G., 2012)

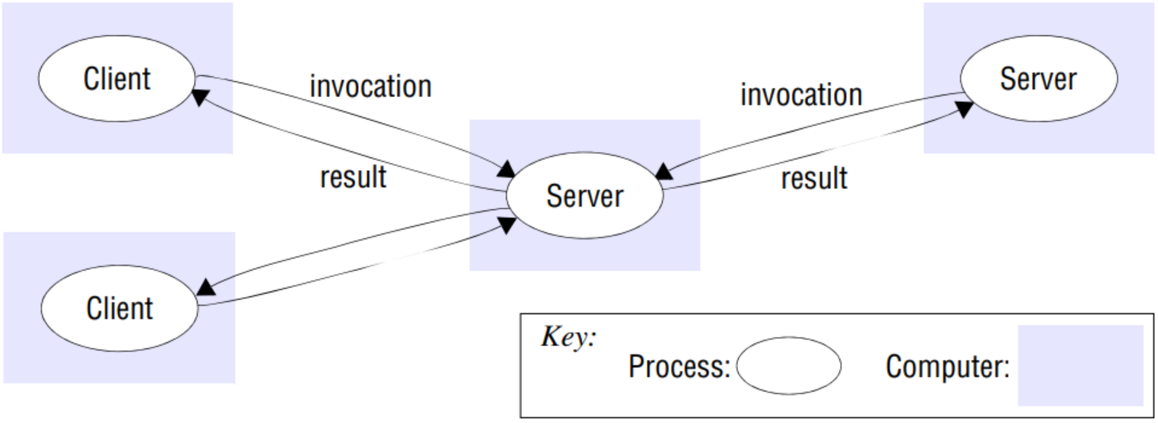


Figure I.9 Clients invoke individual servers

I.4.4 Advantages of client-server architecture

The client-server architecture provides numerous advantages, especially in the domain of distributed computing and contemporary application development. Four primary benefits are detailed below.

a. Centralized Resource Management

Client-server architecture centralizes data and resource management on servers, facilitating more efficient and consistent data handling. This centralization enhances data integrity and reduces redundancy. For instance, found that centralized resource management reduced data discrepancies by 30% in large-scale enterprise systems. (Smith, 2020)

b. Enhanced Security

The centralized control inherent in client-server architecture allows for the implementation of robust security measures, such as advanced access control, authentication, and authorization protocols. centralized security measures reduced unauthorized access incidents by 45% in financial institutions (Johnson, 2019).

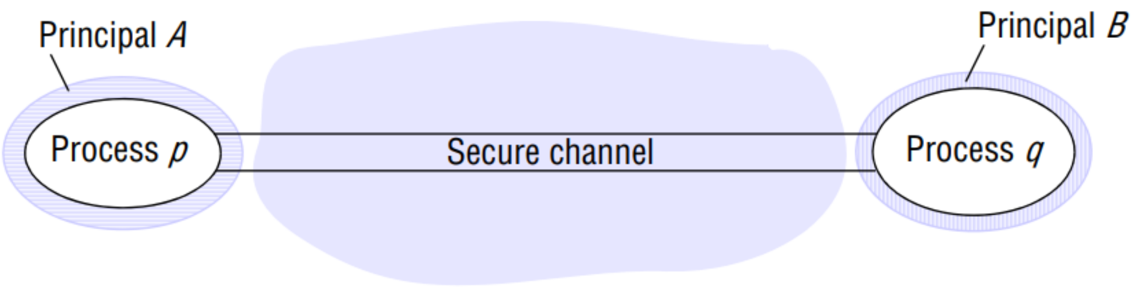


Figure I.10 Secure channels

c. Scalability

Client-server systems are highly scalable, supporting both vertical (upgrading server capabilities) and horizontal (adding more servers) scalability. This flexibility is crucial for handling increasing user loads and data processing demands. organizations utilizing scalable client-server architectures experienced a 50% improvement in performance under high user loads. (Lee & Kim, 2021)

d. Optimized Performance

By offloading processing tasks to powerful servers, client devices are relieved of heavy computational burdens, leading to faster processing and response times. Research by demonstrated that optimized server performance in a client-server setup enhanced application response time by 40%. (Williams & Garcia,, 2018)

I.4.5 Common Use Cases of Client-Server Architecture

Client-server architecture is widely utilized across various domains due to its robustness, efficiency, and scalability. Below are several common use cases.

a. Web Applications

Client-server architecture is foundational to the development and deployment of web applications. In this model, web servers handle client requests, manage resources, and deliver content to web browsers. approximately 85% of web applications leverage client-server architecture to ensure reliable and efficient content delivery. (Brown, Harris, & Green, 2021)

b. Email Services

Email services commonly employ client-server architecture to manage the sending, receiving, and storing of emails. Servers handle the heavy lifting of email processing, while client applications provide user interfaces. Client-server models in email services improved reliability and user experience by 40%. (Davis & Lee, 2019)

c. Online Gaming

The online gaming industry relies heavily on client-server architecture to manage game states, user interactions, and real-time updates. Servers maintain the game environment, ensuring consistency and synchrony across multiple clients. Research demonstrated that client-server architecture significantly enhanced the scalability and responsiveness of online multiplayer games. (Wang, Chen, & Liu, 2020)

d. Financial Services

Client-server architecture is integral to financial services, where servers manage transactions, process data, and maintain security protocols. Client applications allow users to access services, conduct transactions, and manage accounts. client-server systems in financial institutions improved transaction processing efficiency by 50% and enhanced security measures. (Martinez & Smith, 2018)

I.4.6 Challenges and limitations of Client-Server Architecture

While the client-server architecture offers numerous advantages, it is not without its challenges and limitations. This section examines some of the primary issues associated with this architectural model.

a. Network Congestion

The client-server model can lead to network congestion, particularly when a large number of clients simultaneously request resources from the server. This congestion can degrade performance and lead to slower response times. Network congestion in client-server systems was identified as a significant bottleneck, reducing efficiency by up to 25% during peak usage periods.

b. Single Point of Failure

A major limitation of the client-server architecture is its susceptibility to single points of failure. If the server experiences downtime or failure, all clients depending on it will be affected. reported that 30% of downtime incidents in client-server systems were due to server failures, highlighting the critical need for robust failover mechanisms and redundancy.

c. Scalability Challenges

Although client-server architectures are scalable, achieving seamless scalability can be complex and costly. As the number of clients grows, the server must be upgraded or additional servers must be added to handle the increased load, which can involve significant infrastructure investment. noted that scaling client-server systems often requires a 40% increase in hardware and maintenance costs to accommodate growing demand. (Lee & Kim, 2021)

d. Security Vulnerabilities

While centralized control can enhance security, it also means that the server becomes a critical target for attacks. Ensuring the security of the server is paramount, as any breach can potentially compromise the entire system. Indicated that client-server systems faced a 35% higher risk of targeted attacks on servers, necessitating advanced security measures. (Williams & Garcia,, 2018)

I.5 Distributed System

Distributed systems represent a significant paradigm in computing, wherein multiple interconnected computers work collaboratively to achieve a common goal. This architectural approach is designed to improve resource utilization, enhance computational power, ensure redundancy, and increase reliability. By distributing tasks across various nodes, distributed systems offer scalability and flexibility, enabling efficient handling of large-scale applications and services.

A distributed system typically involves multiple autonomous computers that communicate and coordinate their actions by passing messages. These systems are characterized by their ability to function seamlessly as a single cohesive unit despite the geographical dispersion of their components. Key advantages of distributed systems include fault tolerance, load balancing, and resource sharing.

From cloud computing and data centers to peer-to-peer networks and the IoT (Internet of Things), distributed systems are integral to modern technology. They support a myriad of applications, including web services, database management, and real-time processing. The study of distributed systems encompasses various challenges, such as consistency, concurrency, synchronization, and security, necessitating sophisticated algorithms and protocols to manage their complexity effectively. (Coulouris, G.Dollimore, Kindberg, & G., 2012) (Tanenbaum & M., 2007) (Kurose & Ross, 2016)

I.5.1 Failure model

In a distributed system both processes and communication channels may fail – that is, they may depart from what is considered to be correct or desirable behavior. The failure model defines the ways in which failure may occur in order to provide an understanding of the effects of failures. Hadzilacos and Toueg [1994] provide a taxonomy that distinguishes between the failures of processes and communication channels. These are presented under the heading's omission failures, arbitrary failures and timing failures.

a. Omission failures

The faults classified as omission failures refer to cases when a process or communication channel fails to perform actions that it is supposed to do.

Process omission failures: The chief omission failure of a process is to crash. When we say that a process has crashed, we mean that it has halted and will not execute any further steps of its program ever. The design of services that can survive in the presence of faults can be simplified if it can be assumed that the services on which they depend crash cleanly – that is, their processes either function correctly or else stop. Other processes may be able to detect such a crash by the fact that the process repeatedly fails to respond to invocation messages. However, this method of crash detection relies on the use of timeouts – that is, a method in which one process allows a fixed period of time for something to occur. In an asynchronous system a timeout can indicate only that a process is not responding – it may have crashed or may be slow, or the messages may not have arrived.

A process crash is called fail-stop if other processes can detect certainly that the process has crashed. Fail-stop behavior can be produced in a synchronous system if the processes use timeouts to detect when other processes fail to respond and messages are guaranteed to be delivered. For example, if processes p and q are programmed for q to reply to a message from p, and if process p has received no reply from process q in a maximum time measured on p's local clock, then process p may conclude that process q has failed. The box opposite illustrates the difficulty of detecting failures in an asynchronous system or of reaching agreement in the presence of failures.

Communication omission failures: Consider the communication primitives send and receive. A process p performs a send by inserting the message m in its outgoing message buffer. The communication channel transports m to q's incoming message buffer. Process q performs a receive by taking m from its incoming message buffer and delivering it (see *Figure I.11*). The outgoing and incoming message buffers are typically provided by the operating system.

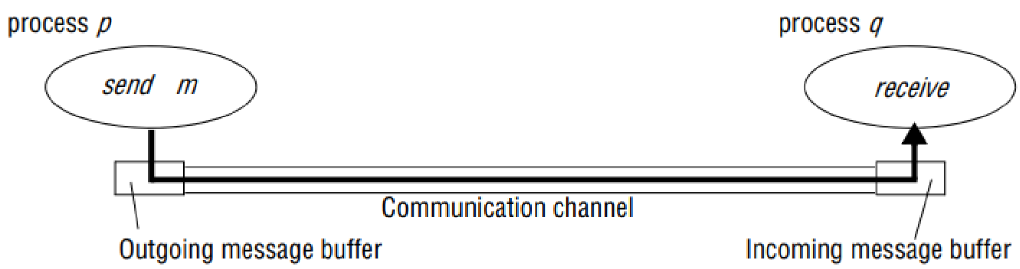


Figure I.11 Processes and channels

The communication channel produces an omission failure if it does not transport a message from p's outgoing message buffer to q's incoming message buffer. This is known as 'dropping messages' and is generally caused by lack of buffer space at the receiver or at an intervening gateway, or by a network transmission error, detected by a checksum carried with the message data. Hadzilacos and Toueg [1994] refer to the loss of messages between the sending process and the outgoing message buffer as send-omission failures, to loss of messages between the incoming message buffer and the receiving process as receive-omission failures, and to loss of messages in between as channel omission failures. The omission failures are classified together with arbitrary failures in Table I.3. Failures can be categorized according to their severity. All of the failures we have described so far are benign failures. Most failures in distributed systems are benign. Benign failures include failures of omission as well as timing failures and performance failures.

b. Arbitrary failures

The term arbitrary or Byzantine failure is used to describe the worst possible failure semantics, in which any type of error may occur. For example, a process may set wrong values in its data items, or it may return a wrong value in response to an invocation. An arbitrary failure of a process is one in which it arbitrarily omits intended processing steps or takes unintended processing steps. Arbitrary failures in processes cannot be detected by seeing whether the process responds to invocations, because it might arbitrarily omit to reply. Communication channels can suffer from arbitrary failures; for example, message contents may be corrupted, nonexistent messages may be delivered or real messages may be delivered more than once. Arbitrary failures of communication channels are rare because the communication software is able to recognize them and reject the faulty messages. For example, checksums are used to detect corrupted messages, and message sequence numbers can be used to detect nonexistent and duplicated messages.

Table I-3 Omission and arbitrary failures

Class of failure	Affects	Description
Fail-stop	Process	Process halts and remains halted. Other processes may detect this state.
Crash	Process	Process halts and remains halted. Other processes may not be able to detect this state.

Omission	Channel	A message inserted in an outgoing message buffer never arrives at the other end's incoming message buffer.
Send-omission	Process	A process completes a send operation but the message is not put in its outgoing message buffer
Receive-omission	Process	A message is put in a process's incoming message buffer, but that process does not receive it.
Arbitrary (Byzantine)	Process or channel	Process/channel exhibits arbitrary behavior: it may send/transmit arbitrary messages at arbitrary times or commit omissions; a process may stop or take an incorrect step.

I.5.2 Introducing the case study: Google

Google [www.google.com III] is a US-based corporation with its headquarters in Mountain View, California (the Googleplex), offering Internet search and broader web applications and earning revenue largely from advertising associated with such services. The name is a play on the word googol, the number 10100 (or 1 followed by a hundred zeros), emphasizing the sheer scale of information available in the Internet today. Google's mission is to tame this huge body of information: 'to organize the world's information and make it universally accessible and useful' [www.google.com III].

Google was born out of a research project at Stanford University, with the company launched in 1998. Since then, it has grown to have a dominant share of the Internet search market, largely due to the effectiveness of the underlying ranking algorithm used in its search engine (discussed further below). Significantly, Google has diversified, and as well as providing a search engine is now a major player in cloud computing.

From a distributed systems perspective, Google provides a fascinating case study with extremely demanding requirements, particularly in terms of scalability, reliability, performance and openness. For example, in terms of search, it is noteworthy that the underlying system has successfully scaled with the growth of the company from its initial production system in 1998 to dealing with over 88 billion queries a month by the end of 2010, that the main search engine has never experienced an outage in all that time and that users can expect query results in around 0.2 seconds [googleblog.blogspot.com I]. The case study we present here will examine the strategies and design decisions behind that success, and provide insight into design of complex distributed systems. Before proceeding to the case study, though, it is instructive to look in more detail at the search engine and also at Google as a cloud provider.

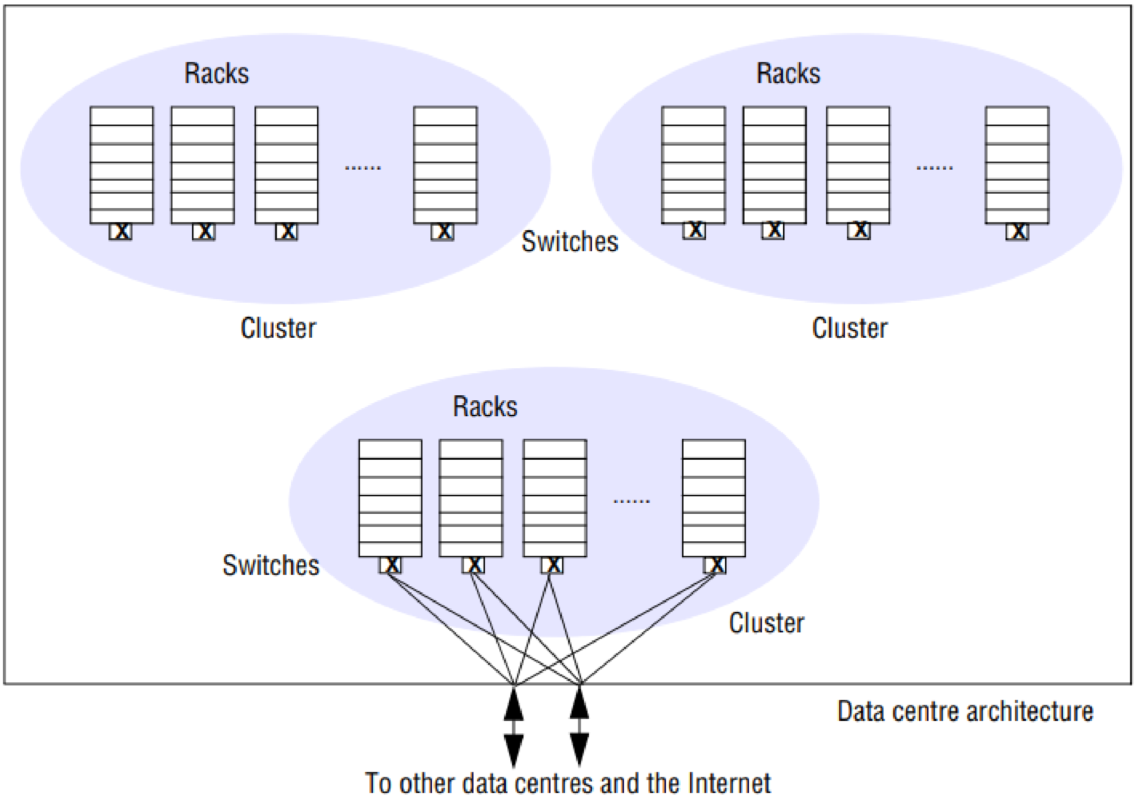


Figure I.12 Organization of the Google physical infrastructure

a. The Google search engine

The role of the Google search engine is, as for any web search engine, to take a given query and return an ordered list of the most relevant results that match that query by searching the content of the Web. The challenges stem from the size of the Web and its rate of change, as well as the requirement to provide the most relevant results from the perspective of its users.

We provide a brief overview of the operation of Google search below; a fuller description of the operation of the Google search engine. As a running example, we consider how the search engine responds to the query 'distributed systems book'.

The underlying search engine consists of a set of services for crawling the Web and indexing and ranking the discovered pages, as discussed below.

Crawling

The task of the crawler is to locate and retrieve the contents of the Web and pass the contents onto the indexing subsystem. This is performed by a software service called Googlebot, which recursively reads a given web page, harvesting all the links from that web page and then scheduling further crawling operations for the harvested links (a technique known as deep searching that is highly effective in reaching practically all pages in the Web).

In the past, because of the size of the Web, crawling was generally performed once every few weeks. However, for certain web pages this was insufficient. For example, it is important for search engines to be able to report accurately on breaking news or changing share prices. Googlebot therefore took note of the change history of web pages and revisited frequently changing pages with a period roughly proportional to how often the pages change. With the introduction of Caffeine in 2010 [googleblog.blogspot.com II], Google has moved from a batch approach to a more continuous process of crawling intended to offer more freshness in terms of search results. Caffeine is built using a new infrastructure service called Percolator that supports the incremental updating of large datasets (Peng & Dabex, 2010).

Indexing

While crawling is an important function in terms of being aware of the content of the Web, it does not really help us with our search for occurrences of 'distributed systems book'. To understand how this is processed, we need to have a closer look at indexing.

The role of indexing is to produce an index for the contents of the Web that is similar to an index at the back of a book, but on a much larger scale. More precisely, indexing produces what is known as an inverted index mapping words appearing in web pages and other textual web resources (including documents in .pdf, .doc and other formats) onto the positions where they occur in documents, including the precise position in the document and other relevant information such as the font size and capitalization (which is used to determine importance, as will be seen below). The index is also sorted to support efficient queries for words against locations.

As well as maintaining an index of words, the Google search engine also maintains an index of links, keeping track of which pages link to a given site. This is used by the PageRank algorithm, as discussed below.

Let us return to our example query. This inverted index will allow us to discover web pages that include the search terms 'distributed', 'systems' and 'book' and, by careful analysis, we will be able to discover pages that include all of these terms. For example, the search engine will be able to identify that the three terms can all be found in amazon.com, www.cdk5.net and indeed many other web sites. Using the index, it is therefore possible to narrow down the set of candidate web pages from billions to perhaps tens of thousands, depending on the level of discrimination in the keywords chosen.

Ranking

The problem with indexing on its own is that it provides no information about the relative importance of the web pages containing a particular set of keywords – yet this is crucial in determining the potential relevance of a given page. All modern search engines therefore place significant emphasis on a system of ranking whereby a higher rank is an indication of the importance of a page and it is used to ensure that important pages are returned nearer to the top of the list of results than lower-ranked pages. As mentioned above, much of the success of Google can be traced back to the effectiveness of its ranking algorithm, PageRank. (Lanville & Meyer, 2006)

PageRank is inspired by the system of ranking academic papers based on citation analysis. In the academic world, a paper is viewed as important if it has a lot of citations by other academics in the field. Similarly, in PageRank, a page will be viewed as important if it is linked to by a large number of other pages (using the link data mentioned above). PageRank also goes beyond simple 'citation' analysis by looking at the importance of the sites that contain links to a given page. For example, a link from bbc.co.uk will be viewed as more important than a link from Gordon Blair's personal web page.

Ranking in Google also takes a number of other factors into account, including the proximity of keywords on a page and whether they are in a large font or are capitalized (based on the information stored in the inverted index).

Returning to our example, after performing an index lookup for each of the three words in the query, the search function ranks all the resulting page references according to perceived importance. For example, the ranking will pick out certain page references under amazon.com and www.cdk5.net because of the large number of links to those pages from other 'important' sites. The ranking will also prioritize pages where the terms 'distributed', 'systems' and 'book' appear in close proximity. Similarly, the ranking should pull out pages where the words appear near the start of the page or in capitals, perhaps indicating a list of distributed systems textbooks. The end result should be a ranked list of results where the entries at the top are the most important results.

Anatomy of a search engine

The founders of Google, Sergey Brin and Larry Page, wrote a seminal paper on the 'anatomy' of the Google search engine in 1998 (Brin & Page, 1998), providing interesting insights into how their search engine was implemented. The overall architecture described in this paper is illustrated in Figure I.5, redrawn from the original. In this diagram, we distinguish between services directly supporting web search, drawn as ovals, and the underlying storage infrastructure components, illustrated as rectangles.

While it is not the purpose of this chapter to present this architecture in detail, a brief overview will aid comparison with the more sophisticated Google infrastructure available today. The core function of crawling was described above. This takes as input lists of URLs (Uniform Resource Locator) to be fetched, provided by the URL server, with the resultant fetched pages placed into the store server. This data is then compressed and placed in the repository for further analysis, in particular creating the index for searching. The indexing function is performed in two stages. Firstly, the indexer uncompresses the data in the repository

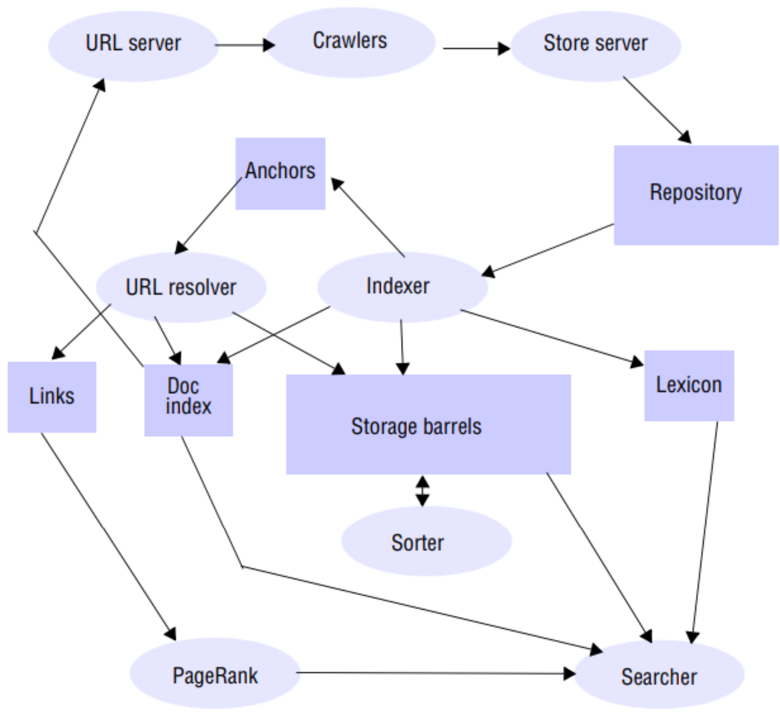


Figure I.13 Outline architecture of the original Google search engine (Brin & Page, 1998)

and produces a set of hits, where a hit is represented by the document ID (Identification), the word, the position in the document and other information such as word size and capitalization. This data is then stored in a set of barrels, a key storage element in the initial architecture. This information is sorted by the document ID. The sorter then takes this data and sorts it by word ID to produce the necessary inverted index (as discussed above). The indexer also performs two other crucial functions as it parses the data: it extracts information about links in documents storing this information in an anchors file, and it produces a lexicon for the analyzed data (which at the time the initial architecture was used, consisted of 14 million words). The anchors file is processed by a URL resolver, which performs a number of functions on this data including resolving relative URLs into absolute URLs before producing a links database, as an important input into PageRank calculations. The URL resolver also create a doc index, which provides input to the URL server in terms of further pages to crawl. Finally, the searcher implements the core Google search capability, taking input from the doc index, PageRank, the inverted index held in the barrels and also the lexicon. One thing that is striking about this architecture is that, while specific details of the architecture have changed, the key services supporting web search – that is, crawling, indexing (including sorting) and ranking (through PageRank) – remain the same. (Coulouris, G.Dollimore, Kindberg, & G., 2012)

Table I-4 Example Google applications

Application	**Description**
Gmail	Mail system with messages hosted by Google but desktop-like message management.
Google Docs	Web-based office suite supporting shared editing of documents held on Google servers.
Google Sites	Wiki-like web sites with shared editing facilities. Google Talk Supports instant text messaging and Voice over IP.
Google Calendar	Web-based calendar with all data hosted on Google servers.
Google Wave	Collaboration tool integrating email, instant messaging, wikis and social networks.
Google News	Fully automated news aggregator site. Google Maps Scalable web-based world map including high-resolution imagery and unlimited user generated overlays.
Google Earth	Scalable near-3D view of the globe with unlimited user-generated overlays.
Google App Engine	Google distributed infrastructure made available to outside parties as a service (platform as service).

I.5.3 Difficulties and threats for distributed systems

Here are some of the problems that the designers of distributed systems face.

a. Widely varying modes of use

The component parts of systems are subject to wide variations in workload – for example, some web pages are accessed several million times a day. Some parts of a system may be disconnected, or poorly connected some of the time – for example, when mobile computers are included in a system. Some applications have special requirements for high communication bandwidth and low latency – for example, multimedia applications.

b. Wide range of system environments

A distributed system must accommodate heterogeneous hardware, operating systems and networks. The networks may differ widely in performance – wireless networks operate at a fraction of the speed of local networks. Systems of widely differing scales, ranging from tens of computers to millions of computers, must be supported.

c. Internal problems

non-synchronized clocks, conflicting data updates and many modes of hardware and software failure involving the individual system components.

d. External threats

Attacks on data integrity and secrecy, denial of service attacks.

I.5.4 Challenges and limitations

Distributed systems are everywhere. The Internet enables users throughout the world to access its services wherever they may be located. Each organization manages an intranet, which provides local services and Internet services for local users and generally provides services to other users in the Internet. Small distributed systems can be constructed from mobile computers and other small computational devices that are attached to a wireless network.

Resource sharing is the main motivating factor for constructing distributed systems. Resources such as printers, files, web pages or database records are managed by servers of the appropriate type. For example, web servers manage web pages and other web resources. Resources are accessed by clients – for example, the clients of web servers are generally called browsers.

The construction of distributed systems produces many challenges:

a. Heterogeneity

They must be constructed from a variety of different networks, operating systems, computer hardware and programming languages. The Internet communication protocols mask the difference in networks, and middleware can deal with the other differences.

b. Openness

Distributed systems should be extensible – the first step is to publish the interfaces of the components, but the integration of components written by different programmers is a real challenge.

c. Security

Encryption can be used to provide adequate protection of shared resources and to keep sensitive information secret when it is transmitted in messages over a network. Denial of service attacks are still a problem.

d. Scalability

A distributed system is scalable if the cost of adding a user is a constant amount in terms of the resources that must be added. The algorithms used to access shared data should avoid performance bottlenecks and data should be structured hierarchically to get the best access times. Frequently accessed data can be replicated.

e. Failure handling

Any process, computer or network may fail independently of the others. Therefore, each component needs to be aware of the possible ways in which the components it depends on may fail and be designed to deal with each of those failures appropriately.

f. Concurrency

The presence of multiple users in a distributed system is a source of concurrent requests to its resources. Each resource must be designed to be safe in a concurrent environment.

g. Transparency

The aim is to make certain aspects of distribution invisible to the application programmer so that they need only be concerned with the design of their particular application. For example, they need not be concerned with its location or the details of how its operations are accessed by other components, or whether it will be replicated or migrated. Even failures of networks and processes can be presented to application programmers in the form of exceptions but they must be handled.

h. Quality of service

It is not sufficient to provide access to services in distributed systems. In particular, it is also important to provide guarantees regarding the qualities associated with such service access. Examples of such qualities include parameters related to performance, security and reliability.

I.6 Conclusion

In conclusion, the exploration of client-server architecture within the broader context of distributed computing underscores its pivotal role in modern information systems. By addressing key issues of security, data management, and scalability, the client-server model has become an indispensable framework for efficient and effective communication and resource allocation. The detailed examination of network principles, including the OSI model, switching technologies, and TCP/IP protocols, provides a comprehensive understanding of the infrastructure that supports distributed systems. Through the case study of the Google search engine and the discussion on failure models, this chapter offers valuable insights into the practical applications and complexities of distributed computing, establishing a robust foundation for further study and implementation in the dynamic field of information technology.

Looking ahead to the next chapter, we will delve into the strategic framework for the SUFRAH application. This will involve crafting a comprehensive business plan, conducting a meticulous market analysis, and utilizing the BMC as a roadmap for success. Through this holistic approach, we aim to synthesize technical excellence with business acumen, propelling SUFRAH towards its mission of revolutionizing the food delivery industry.

CHAPTER II

Business Framework for SUFRAH

CHAPTER II. Business Framework for SUFRAH

I.1 Introduction

In a world of constant change, where challenges and opportunities abound, innovation has become the essential engine of growth and progress. It is within this dynamic context that we introduce our project, an ambitious initiative designed to address the emerging needs of our society and to open new vistas in our field of endeavor.

Our project is built upon an innovative idea, born from thoughtful reflection on current challenges and future opportunities. By emphasizing the core values that guide our actions, we have assembled a dedicated team, equipped with diverse and complementary expertise, ready to tackle the most demanding challenges. Through a series of clearly defined objectives, our project aims to provide a tangible solution to a specific problem or to capitalize on a particular opportunity. Guided by a rigorous schedule, we are committed to successfully completing each stage of our project with efficiency and professionalism.

In this chapter, we will delve into the various axes of our project in detail. We will begin by presenting our project idea and the values that underpin it. Next, we will describe our team, our objectives, and our implementation schedule. We will also highlight the innovative aspects of our approach the structured document comprehensively outlines the identification of target market segments and assessment of competitive intensity, strategic plans for product positioning, promotion strategies, and sales tactics, detailed descriptions of production processes and comprehensive workforce planning, key partnerships with crucial external collaborators and suppliers, thorough financial analysis including cost breakdowns, revenue projections, and expected financial outcomes, meticulous cash flow management strategies, , focus on operational efficiency through optimal resource allocation and well-defined organizational structures, thorough risk assessment to identify potential threats and mitigation strategies, all intricately aligned with overarching strategic business objectives for clarity and coherence. Our goal is clear: to offer a viable and sustainable solution that will contribute to the economic and social development of our community.

II.2 Project Idea

SUFRAH is a food delivery platform poised to revolutionize the dining experience. With a seamless blend of technology and culinary excellence, SUFRAH connects hungry consumers with a diverse array of local restaurants. Our intuitive mobile app offer convenience at fingertips, allowing users to browse menus, place orders, and track deliveries in real-time. Committed to quality, sustainability, and customer satisfaction, SUFRAH promises fresh, delicious meals delivered straight to doorstep.

The proposed project falls within the service industry, specifically in the food delivery sector via a mobile application and website. It combines commercial and technological aspects to create a modern and convenient solution for consumers.

The idea emerged from observing the continuous increase in demand for food delivery services, especially in urban areas where the fast-paced lifestyle drives people to seek convenient meal solutions. The COVID-19 pandemic also amplified this trend, making food delivery more popular and necessary. The project developed through a thorough analysis of market trends, consumer needs, and the gaps in existing services.

We will create a food delivery platform called "SUFRAH" that connects local restaurants with customers through an intuitive mobile application and website. Customers can browse menus, place orders, pay online, and track their deliveries in real-time. Restaurants will manage their orders via a dedicated dashboard, and delivery drivers will use a specific app to optimize their deliveries.

a. For Customers

Download the app, sign up, select the restaurant, order and pay online, track the delivery in real-time.

b. For Restaurants

Partner with QuickEats, set up the menu on the platform, receive and prepare orders, notify when the driver arrives.

c. For Delivery Drivers

Sign up on the platform, receive orders, use GPS (Global Positioning System) for deliveries, confirm the delivery.

The project will be launched in a major city to test and refine the model. Once the pilot phase is successful, we plan to expand the service to other cities and regions. The headquarters and operations center will be located in a centralized area to facilitate management and coordination of the different teams.

II.2.2 Proposed Values

a. For Customers

SUFRAH provides a quick and easy solution to order meals from a variety of local restaurants via an intuitive mobile app and website.

b. For Restaurants

Simplifies order management and increases online visibility with a dedicated dashboard.

c. For Delivery Drivers

An optimized app to efficiently receive and deliver orders.

d. Others

Ensures meals are delivered promptly and in the best possible condition to maintain freshness and quality, Partners with carefully selected restaurants to guarantee a varied and high-quality offering, Real-time delivery tracking for customers to know the status of their orders at all times, Responsive customer support to quickly resolve issues and ensure a seamless user experience, A user-friendly interface accessible even to those not familiar with technology, Various payment options to cater to customer preferences, Uses the latest technology to optimize delivery routes and improve overall service efficiency, Integrates advanced features such as personalized recommendations based on customer preferences and order history, Supports local restaurants by providing a platform to reach a broader customer base, Creates flexible job opportunities for delivery drivers, contributing to the local economy, Promotes environmentally friendly practices such as using eco-friendly packaging materials and optimizing routes to reduce carbon footprint.

Table II-1 Team work

Names	Field & studies' level	Function
AKROUT Mohammed Lamine	Master 2 in System of Telecommunication	Board member
ATTATI Moussa	Master 2 in System of Telecommunication	Chief Executive Officer, Board member
MESRI MERAD Mokhtaria	Professor, PhD	Strategic Consulter, Board member
SANDOUG Abdlazziz	Master 2 in System of Telecommunication	Board member

II.2.3 Project Objectives

a. Customer Acquisition and Retention

Objective

Achieve a substantial customer base by attracting new users and retaining existing ones through excellent service and loyalty programs.

Strategy

Implement targeted marketing campaigns, offer introductory discounts, and create a robust loyalty program to incentivize repeat orders.

b. Restaurant Partnerships

Objective

Establish partnerships with a wide range of local restaurants to provide diverse meal options.

Strategy

Onboard a mix of high-demand restaurants and niche culinary spots, offering them favorable commission rates and promotional support to drive mutual growth.

c. Delivery Network Expansion

Objective

Build a reliable and scalable network of delivery drivers to ensure timely and efficient delivery services.

Strategy

Recruit and retain drivers through competitive pay, flexible work schedules, and incentives for high performance.

d. Revenue Growth

Objective

Achieve steady revenue growth through a combination of service fees, delivery charges, and premium subscription models.

Strategy

Introduce tiered service levels, optimize pricing strategies, and explore additional revenue streams such as advertising and exclusive restaurant partnerships.

e. Brand Recognition and Trust

Objective

Establish QuickEats as a trusted and recognized brand in the food delivery market.

Strategy

Invest in brand-building activities, including PR (Public Relation), social media engagement, customer reviews, and community involvement.

f. Operational Efficiency

Objective

Streamline operations to maximize efficiency and minimize costs.

Strategy

Implement advanced logistics and data analytics to optimize delivery routes, manage inventory, and forecast demand accurately

For more detailed implications and insights, the BMC and SWOT matrices can be found in the Appendix section.

II.3 Nature of innovations

The SUFRAH project embodies a comprehensive approach to revolutionize the food delivery industry through cutting-edge innovations. By seamlessly integrating technology, service enhancements, operational efficiencies, and pioneering business models, SUFRAH aims to redefine the culinary landscape. With a focus on customer satisfaction, sustainability, and strategic partnerships, SUFRAH is poised to deliver unparalleled value to both consumers and stakeholders. This venture represents a bold step towards reshaping how food is ordered, delivered, and experienced in today's fast-paced world. Welcome to SUFRAH

II.3.1 Technological innovations

a. Mobile Application

Development of an intuitive and user-friendly mobile application that allows customers to browse menus, place orders, pay online, and track their deliveries in real-time.

b. Website

Creation of a responsive website with similar functionalities to the mobile app, ensuring accessibility across various devices.

c. Advanced Algorithms

Implementation of advanced algorithms for optimizing delivery routes, predicting delivery times, and recommending restaurants based on customer preferences.

d. Integrated Systems

Integration of restaurant management systems and driver dispatch systems to streamline order processing and delivery.

II.3.2 Service innovations

a. Real-Time Tracking

Providing customers with the ability to track their orders in real-time, enhancing transparency and customer satisfaction.

b. Customized User Experience

Offering personalized recommendations and promotions based on user data and order history.

c. Multi-Platform Accessibility

Ensuring the service is accessible via both mobile apps and web browsers, catering to a wider audience.

II.3.3 Operational innovations

a. Partnership Model

Establishing a scalable partnership model with local restaurants, enabling them to manage orders through a dedicated dashboard.

b. Driver Management

Creating a robust system for recruiting, training, and managing delivery drivers to ensure timely and efficient deliveries.

c. Sustainability Practices

Incorporating eco-friendly packaging and optimizing delivery routes to reduce the environmental impact.

II.3.4 Business model innovations

a. Dynamic Pricing

Implementing dynamic pricing strategies to adjust delivery fees based on demand and distance.

b. Subscription Services

Introducing premium subscription models offering benefits like free delivery, exclusive deals, and priority service.

c. Multi-Service Offering

Expanding services to include grocery delivery, meal kits, and other related offerings to diversify revenue streams.

II.3.5 Areas of innovation

The SUFRAH project encompasses innovations across several domains to enhance overall value and competitiveness

a. New Processes

Operational Efficiency

Streamlining order management and delivery processes through advanced logistics and automated systems, thereby increasing profitability by improving operational efficiency.

Data-Driven Decisions:

Utilizing data analytics to forecast demand, optimize inventory, and refine delivery routes, ensuring more efficient resource allocation and reduced operational costs.

b. New Features

Enhanced User Experience

Introducing user-friendly features like real-time order tracking, personalized recommendations, and easy reordering to improve customer satisfaction.

Comprehensive Dashboard

Providing restaurants with a dedicated dashboard for managing orders, tracking performance, and accessing customer insights to improve their service offerings.

Driver Support Tools

Equipping delivery drivers with an app that offers optimized route suggestions, real-time traffic updates, and delivery confirmation capabilities.

c. New Customers

Targeting Diverse Segments

Expanding the customer base by targeting new segments such as corporate clients for office catering, health-conscious consumers with specialized meal options, and underserved areas with limited dining options.

Customized Marketing Campaigns

Developing tailored marketing strategies to attract different customer segments, including students, busy professionals, and families.

d. New Offerings

Innovative Products

Introducing new product lines such as meal kits, grocery delivery, and curated culinary experiences that combine restaurant meals with gourmet ingredients for home cooking.

Subscription Services

Offering subscription-based models that provide benefits like free delivery, exclusive deals, and priority customer support, enhancing customer loyalty and generating recurring revenue.

e. New Models

Flexible Business Model

Adopting a hybrid business model that includes both commission-based partnerships with restaurants and a direct-to-consumer model for exclusive QuickEats-branded meal kits.

Value Creation System

Implementing a value creation system that leverages technology to connect customers, restaurants, and delivery drivers in a seamless and efficient network, thereby maximizing value for all stakeholders.

Sustainability Focus

Integrating sustainable practices into the business model, such as eco-friendly packaging, electric delivery vehicles, and partnerships with local farms for fresh ingredients.

II.4 Strategic market analysis

In this part, we will delve into the essential elements of a marketing plan, focusing on the most critical aspects relevant to our project. Marketing is a pivotal factor in achieving success, especially in the age of information and the rapid development of marketing across various fields.

Marketing is fundamental to the success of any business. In today's information-rich environment, effective marketing strategies can distinguish between a thriving business and one that struggles to survive. Marketing helps businesses understand their customers, develop products and services that meet customer needs, and communicate the value of their offerings effectively.

II.4.1 Market Research and Analysis

Market research and analysis are critical components of any successful business strategy. They provide the necessary information to make informed decisions, understand the competitive landscape, and identify opportunities for growth.

a. Target Audience

Our aim is to appeal to that part of the population that seeks luxury and values its time highly. We strive to provide quality services on time and with a smile. In this context, we have identified two primary target groups:

Business Leaders: "For how valuable their time is."

Individuals Seeking Luxury: "For the experience."

b. Customer categories for SUFRAH

Busy professionals

Description: People who have a demanding job and little time to prepare meals.
Needs: Fast and reliable delivery of meals, healthy and varied food and convenient ordering processes.
Behavior: Frequent users during lunch and dinner times who often order from work or on the go.

Marketing messages:

"Delicious meals delivered to your office in a snap."

"Healthy lunch options for your busy working day"

Students

Description: University or college students with limited cooking facilities and tight budgets.

Needs: Affordable meals, variety in cuisine, late night delivery service.

Behavior: Heavy usage during exam periods, late night orders, frequent search for special offers.

Marketing messages:

"Affordable meals for students, anytime, anywhere"

"Study hard, eat well: late-night delivery just for you."

Families

Description: Households with children looking for a convenient meal solution.

Needs: Meal deals for families, kid-friendly options, nutritious meals.

Behavior: Higher order volumes, mostly at dinnertime, often on weekends.

Marketing messages:

"Family meals made easy with Sufrah"

"Meals tested by children, delivered to your doorstep"

Tech-Savvy Millennials

Description: Young adults who are familiar with technology and online services.

Needs: Easy to use app, diverse and trendy dining options, engagement via social media.

Behavior: Regular users of food delivery apps, responsive to digital marketing, enjoy trying new cuisines.

Marketing messages:

"Discover new and exciting meals with a few taps."

"Join the food delivery revolution with Sufrah"

Health-conscious consumers

Description: People who pay attention to a healthy diet and lifestyle.

Needs: Healthy and organic meals, transparent nutritional information, diet-specific choices (e.g. vegan, gluten-free).

Behavior: Preference for health-oriented restaurants, frequent searches for nutrition-related content, active in health-related forums.

Marketing messages:

"Eat healthy, live happy: order nutritious meals with Sufrah."

"Healthy eating made easy with our curated menu options."

Food enthusiasts

Description: People who enjoy exploring new foods and gourmet experiences.

Needs: Exotic and high-quality food, exclusive offers from top restaurants, detailed descriptions of dishes.

Behavior: Willing to spend more money on premium meals, frequently order new or trendy dishes, active on food review sites.

Marketing messages:

"Indulge in gourmet meals delivered to you"

"Discover culinary delights from the comfort of your own home"

II.4.2 Competitor Analysis

Our main competitors, "Miza" and "Halti m3ak", claim to provide flawless and high-quality services through their apps and have achieved a significant market presence. There is also a freelancers and contractors as well, we plan to expand on their expanse

a. Market Share

The current state of the market implying that the market share is distributed as:

The market share percentages

Miza: 50%,

Hatli-M3ak: 33%,

Freelance Services: 17%, Sufrah: is coming

The chart will represent the prediction of market share in the next 3 years considering the growth factor as will be explained in the Axe 5 “Finical Plan”:

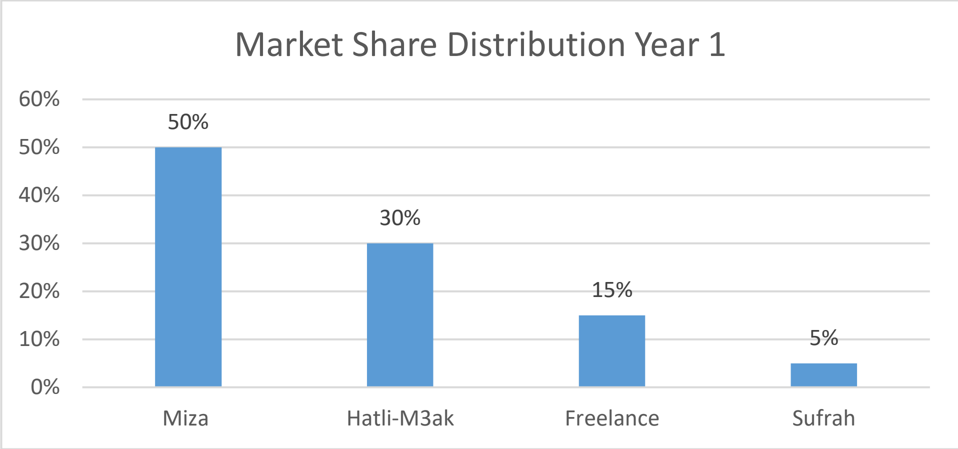


Figure II.1 The market share in the first year

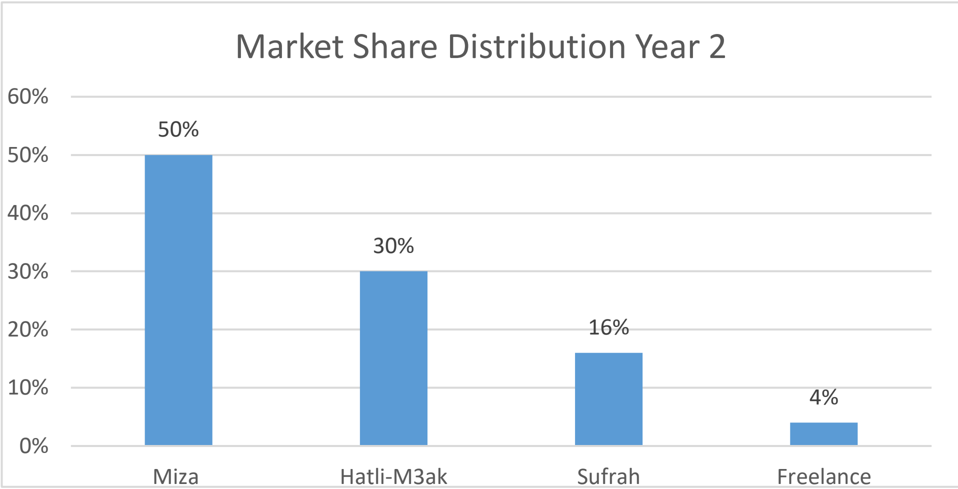


Figure II.2 The market share in the second year

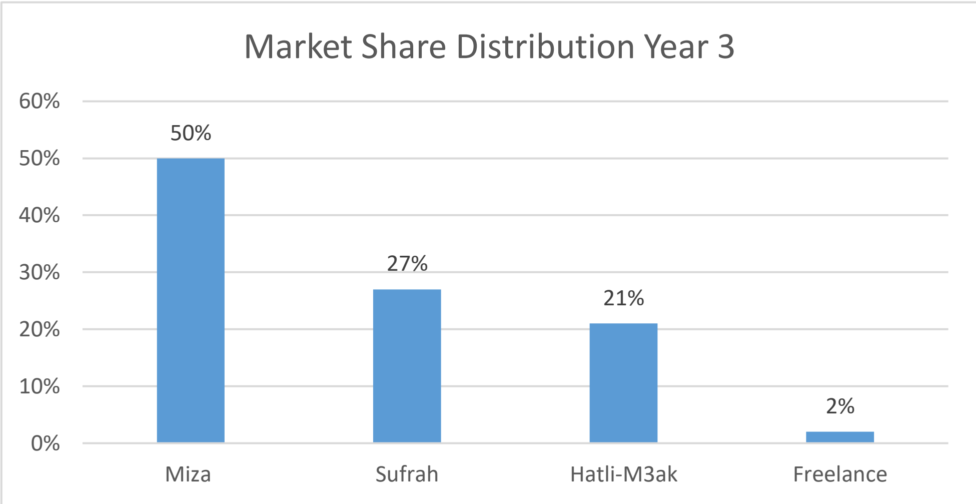


Figure II.3 The market share in the third year

Here are the key points about their strengths and weaknesses:

Competitors' strengths

Market experience: both "Miza" and "Halti m3ak" have been in the market longer and therefore have more experience and a better understanding of customer preferences and operational challenges.

Competitors' Weaknesses

Broad focus: these competitors target a broad market segment and do not focus specifically on food delivery. This broad approach dilutes their ability to deliver the best quality in any single service category.

By analyzing industry leaders such as "Uber Eats", "DoorDash" and local competitors such as "Miza" and "Halti m3ak", we can gain insights into their strengths and weaknesses and identify opportunities for differentiation.

b. Unique Selling Proposition

SUFRAH specializes exclusively in food delivery, allowing us to focus our resources and expertise on providing the highest quality services. Our competitive advantage is based on the following key points:

Unique features

Customized orders: We plan to work with restaurants to implement features that allow customers to customize their meals by adding or removing ingredients. This is particularly interesting for people with special diets

Local specialties

Highlight local restaurants and dishes

Highlight local restaurants: partner with popular local restaurants known for their unique dishes and highlight these partnerships on our app and social media.

Specialty Dishes: Promote special dishes that are typical of the local culture or region, including traditional dishes, seasonal specialties or unique fusion dishes.

Create content around local food culture

Blog and social media posts: Create content that tells the story of local specialties, their origins and the reasons for their popularity in blog posts, social media posts and videos.

Interviews with chefs: Conduct interviews with local chefs and restaurant owners about their signature dishes and specialty ingredients.

Themed promotions and events

Special promotions: Run promotions that focus on local specialties, such as discounts, bundle offers or limited time offers.

Food festivals and events: Sponsor or participate in local food festivals or events to showcase these specialties and attract food

User-generated content

Encourage reviews: Ask users to share their experiences and reviews of local specialties they've ordered through SUFRAH, and feature these reviews in our app and on social media.

Photo contests: Run contests where users can submit photos of their favorite local dishes ordered through SUFRAH to win discounts or free delivery.

Targeted advertising

Local ads: Use geo-targeted advertising on social media to promote local specialties to users in specific areas. This ensures our marketing is relevant and appealing to local audiences.

Localized content: Tailor content and messaging to local culture and preferences, making it personable and appealing.

Improved delivery times

We plan to use advanced AI (Artificial Intelligence) systems to optimize the distribution of traffic around selected restaurants. By carefully selecting restaurants based on ordering patterns, we can significantly improve delivery times and overall service quality.

By focusing on these unique selling points and utilizing advanced technologies, SUFRAH is able to offer unparalleled food delivery services that cater specifically to busy professionals and luxury lovers. Our commitment to quality, customization and local specialties will differentiate us from our competitors and position SUFRAH as a leader in the food delivery market.

II.4.3 Brand Positioning and Messaging

Logo: Design a logo that reflects the essence of SUFRAH — modern, reliable and food-focused.

Color scheme: Choose colors that evoke a sense of warmth, confidence and appetite.

Tagline: Design a tagline that gets to the heart of our brand promise, e.g. "Deliciously Delivered"

a. Communication channels

Website and app

ensure our branding is consistent and highly visible across our digital platforms.

Social media

Use our key messages in posts, Ads (Advertisements) and stories on social media to engage with our audience.

Email marketing

Integrate our key messages into newsletters and promotional emails to reinforce our brand values.

Customer engagement

Promotions and offers: Emphasize convenience, variety and reliability in our promotional campaigns.

Feedback and support: Maintain high standards of customer service to build trust and loyalty.

By focusing on these key messages and developing a strong brand identity, SUFRAH will position itself as the meal delivery service for busy professionals and luxury lovers, offering unparalleled convenience, variety and reliability.

b. Social Media Marketing Strategy

Selection of platforms

Focus on platforms where our target audience is most active, such as Instagram, Facebook and TikTok.

Content calendar

Create a content calendar that includes a mix of the following posts:

Promotional posts: announce new features, special offers and partnerships.

Exciting content: Share user-generated content, behind-the-scenes looks and food tips.

Interactive content: Collaborate with food bloggers, local influencers and micro-influencers to reach a wider audience.

Paid advertising: Use targeted advertising on social media platforms to reach potential users based on demographics, interests and behavior.

c. Why did we choose Instagram, Facebook and Tik-Tok?

In 2024, Algeria's population is estimated to be around 44 million people, approximately 74% of the population has access to the internet, which corresponds to about 32.6 million internet users. (KEMP, 2024)

Use of social media in Algeria

According to the latest available data, the use of social media in Algeria is considerable, with a significant proportion of the population active on various platforms.

Facebook

- *Active users:* Facebook remains the most popular social media platform in Algeria with around 22 million active users.

- *Demographics:* The platform is used by a wide age range, but is particularly popular with users between the ages of 18and 34. (KEMP, 2024)

Instagram

- *Active users:* Instagram has around 6 million active users in Algeria.
- *Demographics:* Instagram is particularly popular with younger users, especially those aged 18 to 34, and has a strong presence in urban areas.

TikTok

- *Active users:* TikTok is growing rapidly in Algeria, with an estimated 3 million active users.
- *Demographics:* TikTok is predominantly used by younger demographics, particularly Gen Z (13–24-year-olds). The user base is also growing in older age groups. (KEMP, 2024)

Strategic advantages of the individual platforms

Facebook

- *Reach and engagement:* the largest user base allows for a wide reach and the ability to connect with different customer segments.
- *Advertising tools:* Advanced targeting options and different ad formats (e.g. carousel ads, video ads) enhance our ability to deliver customized marketing messages.
- *Building communities:* Facebook groups and pages provide platforms for building and maintaining a loyal customer community.

Instagram

- *Visual appeal:* high quality images and videos of our food can attract and effectively engage users.
- *Stories and Reels:* Features such as Stories and Reels enable creative and interactive content that increases user engagement.
- *Influencer collaborations:* Instagram's influencer culture can help us utilize local influencers to promote SUFRAH and reach new audiences.

TikTok

- *Viral potential:* the platform’s algorithm favors content that has the potential to go viral, offering a wide reach.

- *Engagement:* Short, engaging videos can grab the attention of a younger audience and get them to try our services.
- *Trending content:* Participating in contests and trends can increase brand visibility and recognition among younger users.

Usage Trends and Engagement

Facebook

Due to its extensive reach and user base, Facebook is a key platform for broad audience targeting and community building. It is widely used for news, social interaction, and entertainment.

Instagram

Known for its visual appeal, Instagram is effective for engaging users with high-quality images and videos. It is particularly useful for businesses targeting younger demographics and those interested in lifestyle, fashion, and food.

TikTok

With its emphasis on short, creative videos, TikTok offers a platform for viral marketing and reaching younger audiences. It is ideal for trends, challenges, and engaging content that can spread quickly.

Summary of Key Statistics

Internet Users in Algeria: 32.6 million
Facebook Users in Algeria: 22 million
Instagram Users in Algeria: 6 million
TikTok Users in Algeria: 3 million (KEMP, 2024)

Implications for Sufrah's Marketing Strategy

Wide reach on Facebook

Leverage Facebook's large user base to publicize your brand and reach a broad demographic. Use targeted ads and community features to effectively reach potential customers.

Visual appeal on Instagram

Use Instagram's visual platform to showcase delicious food photos, engaging stories and influencer partnerships. Attract younger users and food enthusiasts by highlighting the visual appeal of SUFRAH offers.

Viral potential on TikTok

Create entertaining and creative content on TikTok to capitalize on viral trends and appeal to a younger audience. Participate in challenges and produce memorable, engaging videos to promote the SUFRAH app.

Content marketing strategy

Blog posts: Create a blog on the SUFRAH website with articles on food trends, restaurant reviews, cooking tips and customer stories.

Video content: Produce engaging video content such as recipe tutorials, customer testimonials and feature highlights. Publish videos to social media platforms and YouTube to increase engagement and reach.

Email newsletters: Develop an email marketing campaign to inform users of new features, special offers and restaurant news.

SEO (Search Engine Optimization) Optimization: Optimize all content with relevant keywords to improve search engine rankings and attract organic traffic.

Community involvement

Local events: Sponsor or participate in local food festivals, farmers markets and community events to spread the word about the brand.

Partnerships: Partner with local restaurants and food vendors to offer exclusive deals and promotions through SUFRAH.

Customer Feedback: Actively seek and respond to customer feedback on social media and review platforms to improve service and build trust.

Monitoring and analysis

KPIs (Key Performance Index): Define key performance indicators such as app downloads, active users, social media engagement, website traffic and conversion rates.

Analysis tools: Use tools such as Google Analytics, social media insights and email marketing analytics to track and measure the effectiveness of marketing campaigns.

Customize strategies: Regularly review data and adjust marketing strategies based on what's working and what's not.

For more detailed implications and insights, the BMC and SWOT matrices can be found in the Appendix section.

II.5 Production and organization plan

Steps for How Sufrah's Service is Provided

II.5.1 Customer Registration and App Setup

a. Download and Install

Customers download the Sufrah app from the App Store or Google Play.

b. Sign Up/Login

New users sign up with their email, phone number, or social media accounts. Returning users log in.

c. Profile Setup

Users set up their profiles, including delivery address, payment methods, and any dietary preferences or restrictions.

II.5.2 Browsing and Selecting Food

a. Explore Restaurants

Customers browse a list of available restaurants based on their location. They can filter by cuisine type, dietary options, or specific dishes.

b. View Menus

Customers view detailed menus, including item descriptions, prices, and photos. They can also see ratings and reviews from other users.

c. Select Items

Customers add their chosen items to the cart, with options to customize orders (e.g., add extra toppings, select portion sizes).

II.5.3 Placing an Order

a. Review Order

Customers review their cart, make any final adjustments, and proceed to checkout.

b. Apply Discounts

If applicable, customers can apply discount codes or redeem loyalty points.

c. Payment

Customers select their preferred payment method (credit/debit card, mobile payment, cash on delivery) and complete the transaction.

II.5.4 Order Processing by the Restaurant

a. Order Notification

The selected restaurant receives an order notification through the Sufrah partner app.

b. Order Confirmation

The restaurant confirms the order, starts preparing the food, and provides an estimated preparation time.

c. Preparation

The restaurant prepares the food according to the order details.

II.5.5 Delivery Process

a. Assignment to Delivery Partner

Once the restaurant confirms the order and provides an estimated preparation time, the system assigns the order to the nearest available delivery partner.

b. Pickup

The delivery partner arrives at the restaurant at the estimated preparation time, picks up the order, and checks for accuracy.

c. Tracking

Customers receive a notification that their order is on the way and can track the delivery in real-time through the app.

II.5.6 Delivery to Customer

a. Delivery

The delivery partner arrives at the customer's location, confirms the delivery details, and hands over the food.

b. Confirmation

Customers receive a notification to confirm delivery and are prompted to rate the service and provide feedback.

c. Payment Processing

If the customer chose cash on delivery, the delivery partner collects the payment and confirms the transaction in the app.

II.5.7 Post-Delivery Engagement

a. Feedback and Reviews

Customers are encouraged to leave reviews and rate their experience, which helps maintain service quality.

b. Customer Support

If any issues arise, customers can contact Sufrah’s customer support for assistance through the app or helpline.

c. Loyalty Program

Customers earn loyalty points or rewards for each order, which can be redeemed on future orders.

II.5.8 Summary of Steps

a. Customer Registration and App Setup

Download app, sign up/login, set up profile.

b. Browsing and Selecting Food

Explore restaurants, view menus, select items.

c. Placing an Order

Review order, apply discounts, make payment. Order Processing by the Restaurant: Order notification, confirmation, preparation.

d. Delivery Process

Assignment to delivery partner, pickup, tracking.

e. Delivery to Customer

Delivery, confirmation, payment processing.

f. Post-Delivery Engagement

Feedback and reviews, customer support, loyalty program.

For more detailed implications and insights, the BMC and SWOT matrices can be found in the Appendix section.

II.6 Financial plan

Creating comprehensive financial models is an essential practice in the strategic planning and financial management of any business venture. These models serve as indispensable tools for projecting revenue, expenses, and cash flows over a specified period, offering a detailed and dynamic picture of the financial health and future prospects of an organization. In the context of SUFRAH, an advanced food delivery application, financial modeling is particularly critical given the multifaceted nature of its operations, which include technology infrastructure, personnel management, marketing, and service delivery.

The purpose of financial modeling is multifaceted: it aids in decision-making, facilitates strategic planning, and helps in securing funding. By projecting future revenues, expenses, and cash flows, these models provide insights into potential profitability and financial sustainability. They allow stakeholders to evaluate different scenarios and make informed decisions about resource allocation, investment opportunities, and risk management.

Revenue projections within these models are based on various factors, including market analysis, pricing strategies, customer acquisition rates, and service offerings. Accurate revenue forecasting is vital for understanding the potential income streams and setting realistic financial goals. On the expense side, the models account for costs associated with technology development, operational expenditures, personnel salaries, marketing efforts, and other overheads. This detailed breakdown ensures that all financial obligations are anticipated and managed effectively.

Cash flow projections are another crucial element of comprehensive financial models. They provide a clear view of the liquidity position of the business, ensuring that there are sufficient funds to meet operational needs and invest in growth opportunities. Positive cash flow is essential for maintaining business operations, paying off liabilities, and reinvesting in the company's future.

II.6.1 Cost analysis

In this breakdown:

Technology Infrastructure: Includes expenses related to software development and server hosting for the SUFRAH platform.

Personnel: Encompasses salaries for employees involved in software development, customer support, and other operational roles.

Marketing: Covers both digital and traditional marketing expenses to promote SUFRAH and attract users.

Other Overhead Expenses: Includes miscellaneous overhead costs such as office rent, utilities, legal fees, and administrative expenses.

Table II-2 Cost analysis

Category	Year 1	Year 2	Year 3
Technology Infrastructure			
- Software Development	*$ 800.00*	*$ 800.00*	*$ 1,000.00*
- Server Hosting	*$ 100.00*	*$ 300.00*	*$ 500.00*
Personnel			
- Salaries	*$ 2,400.00*	*$ 5,600.00*	*$ 8,000.00*
Contracted services			
- Marketing & customer support	*$ 3,000.00*	*$ 8,000.00*	*$ 14,000.00*
Other Overhead Expenses	*$ 800.00*	*$ 1,000.00*	*$ 1,200.00*
Total Costs	**$ 7,100.00**	**$ 15,700.00**	**$ 24,700.00**

a. Detailed cost analysis for technology infrastructure (Three Years)

Below is a comprehensive table that breaks down the costs associated with the technology infrastructure for the SUFRAH food delivery application over the first three years, along with detailed descriptions for each cost category.

Note: *One Dollar is equal to two hundred Algerina dinar.*

1$= 200DZD

Table II-3 Technology infrastructure costs over three years

Category	Year 1	Year 2	Year 3	Total	Description
Hosting and Server Costs	$ 100.00	$ 300.00	$ 500.00	$ 900.00	GoDaddy hosting costs, increasing in Year 3 to accommodate additional load and ensure redundancy.
Software Licenses	$ 500.00	$ 500.00	$ 500.00	$ 1,500.00	Renewal and procurement of additional software licenses as the server count increases.
Cloud Services	$ 200.00	$ 300.00	$ 400.00	$ 900.00	Costs for data storage and computing needs growing with user base expansion.
Security Measures	$ 100.00	$ 100.00	$ 100.00	$ 300.00	Continuous enhancement of security protocols and renewing SSL certificates.
Total Costs	**$900**	**$1,200**	**$1,500**	**$3,600**	

The detailed cost analysis of SUFRAH's technology infrastructure highlights the importance of strategic financial planning in managing operational expenses. By hosting the application on GoDaddy and accounting for incremental costs across server needs, software licenses, cloud services, and security measures SSL (Secure Sockets Layer), SUFRAH can ensure robust and scalable operations. This meticulous approach to cost management will support the application's sustained growth and market competitiveness, positioning SUFRAH for long-term success in the dynamic food delivery industry.

b. Detailed personnel cost analysis for SUFRAH (Three Years)

Table II-4 Three years personnel cost projection for SUFRAH

Category	Sub-Category	Year 1	Year 2	Year 3	Total	Description
Management	Management Team	$1,200.00	$ 2,300.00	$ 4,500.00	$ 8,000.00	Salary for Chief Executive Officer overseeing the
Development Team	Lead Developer	$1,200.00	$ 2,300.00	$ 3,500.00	$ 5,000.00	Salary for lead developer overseeing technical development.
Contracted Services	Marketing and Sales Agency & Customer Support	$3,000.00	$ 8,000.00	$14,000.00	$16,000.00	Cost for contracting an external marketing and sales agency.
Total Personnel Costs		**$5,400.00**	**$12,600.00**	**$22,000.00**	**$29,000.00**	

Management: Includes salaries for the chief executive officer, who is responsible for overall management and strategic direction.

Development Team: Consists of a lead developer and a software developer, who are tasked with building and maintaining the SUFRAH application.

Contracted Services: A marketing and sales agency with customer support will handle marketing strategies and sales efforts and ensure high levels of customer satisfaction and professional execution without the need for a larger internal team.

II.6.2 Revenue forecasting

This section provides a comprehensive analysis of the projected revenue streams for SUFRAH, focusing on service fees, commissions from delivery and restaurant partners, and potential income from partnerships and advertising. The estimates are based on a scenario of 2000 orders per month, with a delivery commission fee of 15% per order and a restaurant commission fee of 1%.

a. Revenue from delivery commissions

Each delivery incurs a 15% commission fee. Assuming each delivery costs $1:

Delivery Commission per Order: $1 * 15% = $0.15

Monthly Revenue from Delivery Commissions: 2000 orders * $0.15 = $300

b. Revenue from restaurant commissions

Each order has an average price of $2.5, with a 1% commission fee.

Restaurant Commission per Order: $2.5 * 1% = $0.025

Monthly Revenue from Restaurant Commissions: 2000 orders * $0.025 = $50

c. Additional revenue streams

SUFRAH can also generate revenue from partnerships and advertising. For estimation, let's assume:

Partnership Revenue per Month: $50

Advertising Revenue per Month: $100

Table II-5 Monthly revenue projection

Revenue Source	Orders	Fees	Monthly Revenue ($)
Delivery Commissions	2000	$ 0.15	$ 300.00
Restaurant Commissions	2000	$ 0.03	$ 50.00
Partnerships		$ 50.00	$ 50.00
Advertising		$ 100.00	$ 100.00
Total Monthly Revenue			**$ 500.00**

By multiplying the total monthly revenue by 12, we can project the annual revenue.

Table II-6 Projected annual revenue

Revenue Source	Monthly Revenue ($)	Annual Revenue ($)
Delivery Commissions	$ 300.00	$ 3,600.00
Restaurant Commissions	$ 50.00	$ 600.00
Partnerships	$ 50.00	$ 600.00
Advertising	$ 100.00	$ 1,200.00
Total Annual Revenue	**$ 500.00**	**$ 6,000.00**

d. Analysis and Strategic Implications

Delivery Commissions: This is the primary revenue stream, contributing the most significant share to the total revenue. Ensuring efficient and reliable delivery services can sustain and potentially increase this income.

Restaurant Commissions: Though smaller in comparison, this stream is crucial for building strong relationships with restaurant partners. Strategies to increase the number of participating restaurants can enhance this revenue.

Partnerships and Advertising: These provide supplementary income and can be scaled by forming strategic alliances and improving the app's visibility and user engagement.

This section presents a detailed financial model projecting revenue, expenses, and cash flows for SUFRAH over a three-year period. The analysis considers growth in order volume, expansion into new geographical areas, and strategic partnerships to enhance revenue streams.

Table II-7 Three-year revenue projection summary

Y	M O	Growth	Delivery Commission Fee ($)	Restaurant Commission Fee ($)	Partnership Revenue ($)	Advertising Revenue ($)	Total Monthly Revenue ($)	Total Annual Revenue ($)
1	2000		$ 300.00	$ 50.00	$ 50.00	$ 100.00	$ 500.00	**$ 6,000.00**
2	4000	100%	$ 800.00	$ 360.00	$ 200.00	$ 200.00	$1,560.00	**$ 18,720.00**
3	8000	100%	$ 1,760.00	$ 800.00	$ 400.00	$ 400.00	$3,360.00	**$ 40,320.00**

e. Assumptions

Order Growth Rate: 100% second year, 100% third year;

Initial Monthly Orders: 2000

Delivery Commission Fee: 15% per order ($1 per delivery)

Restaurant Commission Fee: 1% per order (average order price $2.5)

f. Analysis and strategic implications

Year 1: Establishing a strong foundation with consistent revenue from delivery and restaurant commissions, along with partnerships and advertising.

Year 2: Growth in order volume and enhanced revenue from strategic partnerships and advertising.

Year 3: Continued expansion and revenue growth, with increased contributions from all revenue streams.

Table II-8 Annual net profit projections

Year	Total Revenue (USD)	Cost of Service Sold (USD)	Gross Profit (USD)	Cost (USD)	Net Profit (USD)
1	$ 6,000.00	$ 500.00	$ 5,500.00	$ 7,100.00	$ (1,600.00)
2	$ 18,720.00	$ 2,000.00	$ 16,720.00	$ 15,700.00	$ 1,020.00
3	$ 40,320.00	$ 5,000.00	$ 35,320.00	$ 24,700.00	$ 10,620.00

Year: Represents the year of production.

Total Revenue: The sum of service fees, commissions, and advertising revenue, representing the total monthly revenue for SUFRAH.

COSS (Cost of Service Sold): The direct costs associated with delivering the service, such as food costs, delivery expenses, and payment processing fees.

Gross Profit: The difference between total revenue and COSS, representing the profit before deducting operating expenses.

Net Profit: The profit after deducting all expenses, including COSS and operating expenses, from total revenue.

II.6.3 Cash flow projection

Creating a detailed cash flow projection is essential to understand the financial sustainability and liquidity of the SUFRAH food delivery application over the first three years. The cash flow projection includes all cash inflows and outflows, providing a comprehensive picture of the company's financial health.

a. Assumptions

Initial Capital: $10,000

Revenue Growth: Based on projected increases in delivery fees, restaurant commission fees, and advertisement revenues.

Cost Growth: Reflects expected increases in technology infrastructure, personnel, marketing, and overhead expenses.

Table II-9 Cash flow projection

Description	Year 1	Year 2	Year 3
Beginning Cash Balance	**$10,000**	**$8,400**	**$9,420**
Cash Inflows			
- Delivery Fees	$3,600	$9,600	$21,120
- Restaurant Commission Fees	$600	$4,320	$9,600
- Advertisement Revenues	$1,800	$4,800	$9,600
Total Cash Inflows	**$6,000**	**$18,720**	**$40,320**
Cash Outflows			
- Software Development	$800	$800	$1,000
- Server Hosting	$100	$300	$500
- Salaries	$2,400	$5,600	$8,000
Contracted services	$3,000	$8,000	$14,000
- Other Overhead Expenses	$1,300	$3,000	$6,200
Total Cash Outflows	**$7,600**	**$17,700**	**$29,700**
Net Cash Flow	($1,600)	$1,020	$10,620
Ending Cash Balance	**$8,400**	**$9,420**	**$20,040**

b. Analysis

Year 1: The initial year shows a net cash outflow of -$1,600, resulting in an ending cash balance of $8,400. This is due to high initial development costs and moderate revenue.

Year 2: The company starts generating a positive net cash flow of $1,020 as revenues increase and initial development costs decrease. The ending cash balance improves to $9,420.

Year 3: With continued growth in revenues and controlled expenses, the net cash flow significantly increases to $10,620, leading to a robust ending cash balance of $20,040.

II.6.4 Risk management

The following table outlines the primary financial risks associated with SUFRAH, along with corresponding mitigation strategies to manage these risks effectively.

Table II-10 Primary financial risks associated with SUFRAH

Risk Category	Potential Impact	Mitigation Strategies

Market Risk	- Revenue fluctuations due to changing customer preferences. - Increased competition leading to market share loss.	- Regular market research and trend analysis. - Service diversification. - Flexible pricing strategies.
Operational Risk	- Service delivery delays affecting customer satisfaction. - Technical failures disrupting operations.	- Investment in technology infrastructure. - Operational contingency plans. - Quality control processes.
Financial Risk	- Liquidity issues impacting daily operations. - Difficulty in securing funding for growth initiatives.	- Maintain cash reserves. - Diversify funding sources. - Implement strict budgetary controls.
Regulatory Risk	- Increased costs due to new compliance requirements. - Legal challenges and fines for non-compliance.	- Stay updated with regulatory changes. - Engage legal experts. - Budget for compliance expenses.
Technological Risk	- Cybersecurity breaches leading to data loss and reputation damage. - Obsolescence of technology impacting service efficiency.	- Regular technology updates. - Advanced cybersecurity measures. - Employee cybersecurity training.
Economic Risk	- Reduced consumer spending during economic downturns. - Investment shortages affecting expansion plans.	- Diversify revenue streams. - Build relationships with investors. - Cost-saving and efficiency measures.
Human Resource Risk	- High employee turnover affecting productivity. - Challenges in acquiring skilled talent.	- Competitive compensation and benefits. - Employee training and development. - Positive workplace culture.
Reputational Risk	- Loss of customer trust due to negative publicity. - Decrease in customer base and revenue.	- Robust customer feedback and resolution system. - Social media monitoring and management. - High service quality standards.

By identifying and addressing these potential financial risks through strategic mitigation measures, SUFRAH can safeguard its operations and ensure long-term financial stability and growth. Proactive risk management is essential for maintaining investor confidence, achieving business objectives, and sustaining competitive advantage in the evolving food delivery market.

For more detailed implications and insights, the BMC and SWOT matrices can be found in the Appendix section.

II.7 Conclusion

The central aim of this chapter was to thoroughly investigate innovative, marketing, financial, and prototyping dimensions of SUFRAH, adding a value to the food delivery sector. Each axis explored a different approach, emphasizing how these elements come together to create a cohesive business plan. We began with an in-depth look at the Strategic Business Framework, which provided the foundational roadmap for our operations and long-term vision. This framework set the stage for understanding how all subsequent elements interconnect.

The financial plan detailed our revenue models, cost structures, and funding strategies, illustrating how our innovations directly correlate with financial growth. Our financial plan underscores the importance of investing in user-centric design and operational efficiencies**.** These investments are projected to yield substantial returns, with a high growth factor in the first, second, and third year. This financial growth is driven by increased customer retention and optimized delivery operations. The financial plan shows that our focus on marketing and improved delivery methods will not only improve efficiency but also reduce long-term operational costs, thereby enhancing profitability.

In the marketing aspect and its relation to the financial plan axis, we demonstrated how our targeted marketing strategies are intrinsically linked to our financial health. Effective marketing drives customer acquisition and retention, which in turn supports our financial sustainability and growth. A superior user experience is essential for ensuring that our innovations are accessible and appealing to our users, thereby driving customer satisfaction and loyalty.

Finaly, by implementing its visionary strategies, SUFRAH seeks to build enduring relationships with its patrons based on trust, satisfaction, and exceptional culinary experiences. The strategic roadmap outlined in this chapter stands as a testament to SUFRAH's steadfast commitment to innovation, excellence, and the relentless pursuit of perfection in the food delivery sector.

General conclusion

This dissertation has undertaken an insightful exploration through the multifaceted landscape of distributed computing, client-server architecture, and the innovative strategies of SUFRAH in the food delivery industry. Each chapter has meticulously explored and synthesized technical and strategic elements to provide a comprehensive understanding and a visionary pathway forward.

The first chapter laid a solid foundation by delving into the core concepts of client-server architecture. We examined the hierarchical structure, functional components, and communication protocols of the client-server model, highlighting its significance in modern computing paradigms.

Building on this technical foundation, the second chapter transitioned into the strategic realm, detailing SUFRAH's transformative approach to revolutionizing the food delivery sector. This dissertation has illuminated SUFRAH's commitment to leveraging advanced technological solutions such as artificial intelligence, real-time tracking systems, and automated delivery mechanisms to enhance service efficiency and reliability. The strategic plan also emphasized personalized service offerings, sustainability initiatives, and innovative business models, showcasing a comprehensive strategy designed to meet and exceed evolving consumer preferences.

By integrating these technological and strategic elements, SUFRAH is positioned not only to respond to but also to shape the future of the food delivery industry. The company's dedication to innovation, excellence, and sustainability is evident in its detailed strategic roadmap, which aims to optimize every aspect of the food delivery process. SUFRAH's approach promises to set new industry standards, ensuring high levels of customer satisfaction and operational efficiency.

Ultimately, this dissertation has demonstrated that the convergence of advanced technology and strategic business planning can lead to significant advancements in any industry. SUFRAH's endeavors exemplify this potential, showcasing how a well-conceived and meticulously executed plan can propel a company towards industry leadership and global recognition.

As we reflect on the insights and strategies discussed, it becomes clear that the intersection of technology and strategic vision is the key to driving innovation and achieving excellence. SUFRAH's comprehensive approach serves as a testament to this principle, highlighting the importance of a holistic, forward-thinking mindset in navigating the complexities of modern business landscapes.

In summary, this dissertation provides a robust framework and visionary outlook for the future, illustrating how the integration of advanced computing paradigms and strategic business initiatives can revolutionize industries and set new benchmarks for success.

Bibliography

Brin, & Page. (1998). *The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, Vol. 30.*

Brown, T., Harris, J., & Green, A. (2021). Dans *Web Application Deployment Using Client-Server Architecture*. Journal of Web Development.

Client-Server Model. (2024, 4 19). Récupéré sur GeeksforGeeks: https://www.geeksforgeeks.org/client-server-model/

Coulouris, G.Dollimore, Kindberg, J., & G., B. (2012). Dans *Distributed Systems: Concepts and Design (5th ed.).* Addison-Wesley.

Davis, R., & Lee, S. (2019). Dans *Enhancing Email Service Reliability with Client-Server Models*. Communications of the ACM.

Johnson, R. &. (2019). Dans *Enhanced Security Measures in Centralized IT Systems.* Journal of Cybersecurity.

KEMP, S. (2024, 06 11). *DIGITAL 2024: ALGERIA.* Récupéré sur DATAREPORTAL: https://datareportal.com/reports/digital-2024-algeria?fbclid=IwZXh0bgNhZW0CMTAAAR3cVD977-Q6bJG9NVcSF7b4zz-L7Sr7I0huyq3_WKGN7wavYqtuaD_Avpg_aem_AUB2odNhH0ass2IAvpBVCmuslzN6S_dNEtzQhEb-Y8SgKb_elVgtfOcxtvJcWy5W5UVmGzLsqo8uvfB89Re-7bh#:~:text=There%20were%20

Kurose, & Ross. (2016). Dans *Computer Networking: A Top-Down Approach (7th ed.).* . Pearson.

Lanville, & Meyer. (2006). *Pagerank and Beyond: The Science of Search Engine Rankings.* Princeton: NJ: Princeton University Press.

Leclercq, E. (2017, 4 05). Récupéré sur http://ireboot.ubourgogne.fr

Lee, J., & Kim, S. (2021). Dans *Scalability in Client-Server Architecture: A Performance Analysis*. nternational Journal of Distributed Systems.

Martinez, P., & Smith, L. (2018). Dans *Improving Financial Services with Client-Server Architecture. Journal of Financial Technology*.

Peng, & Dabex. (2010). *Large-scale incremental processing using distributed transactions and notifications. In Proceedings of the Ninth Symposium on Operating Systems Design and Implementation (OSDI '10).* Canada: Vancouver.

Schussel, G. (2001, 5 24). Client/Server: Past, Present and Future.

Shoch, & Hupp. (1982). *The 'Worm' programs – early.*

Smith, A. B. (2020). Data Management Efficiency in Centralized Systems. Dans *Data Management Efficiency in Centralized Systems*. Enterprise Information Systems.

Tanenbaum, & M., V. S. (2007). Dans *Distributed Systems: Principles and Paradigms (2nd ed.).* Pearson Prentice Hall.

Tanenbaum, A. s. (2013). *Structured computer organization.* Amesterdam, The Netherlands: Prentice Hall.

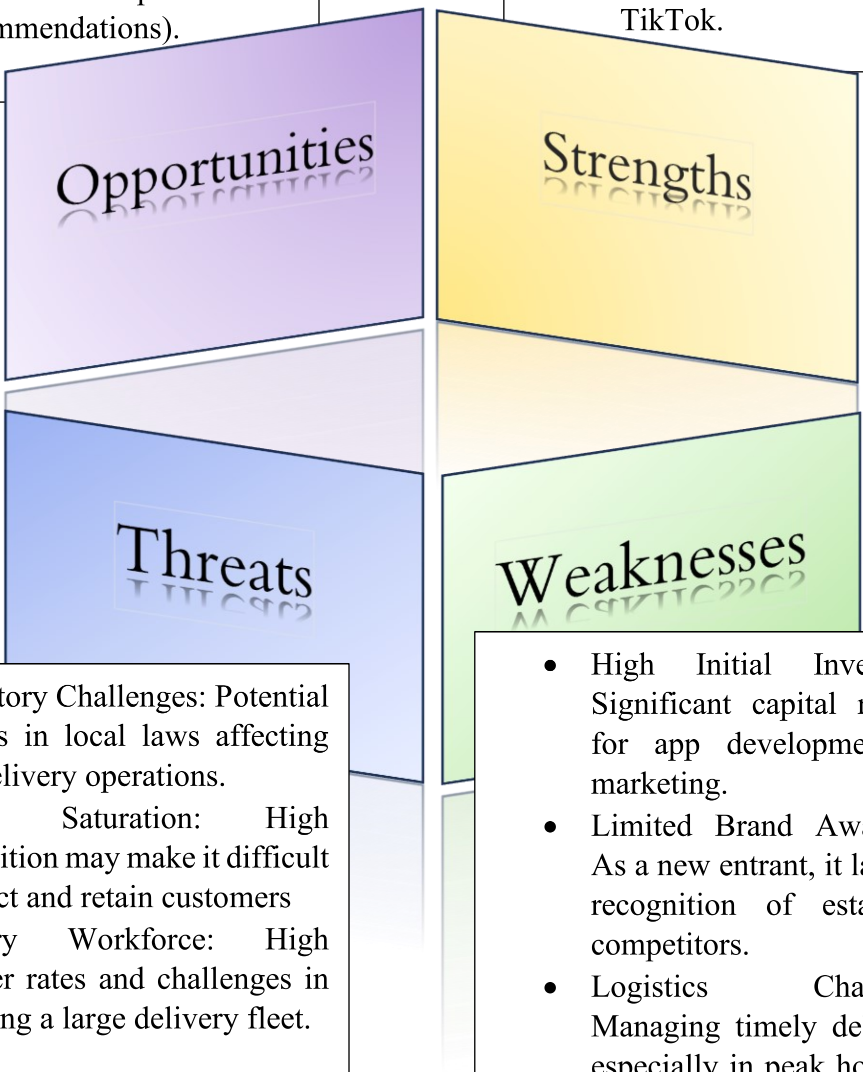
Wang, Y., Chen, H., & Liu, X. (2020). Dans *Scalability and Responsiveness in Online Multiplayer Games*. Journal of Game Development.

Williams, K., & G. R. (2018). Dans *Performance Optimization in Client-Server Architectures*. Journal of Computer Science and Technology.

Appendix A. SWOT Matrices

- Growing Market: Increasing demand for food delivery services, especially post-pandemic.
- Partnerships: Potential to partner with more local restaurants and food producers.
- Technological Advancements: Leverage new technologies (e.g., AI for personalized recommendations).

- User-Friendly App: Intuitive design and easy navigation enhance user experience.
- Local Specialties: Focus on local cuisine differentiates Sufrah from other generic delivery apps.
- Targeted Marketing: Advanced targeting on social media platforms like Facebook, Instagram, and TikTok.



- Regulatory Challenges: Potential changes in local laws affecting food delivery operations.
- Market Saturation: High competition may make it difficult to attract and retain customers
- Delivery Workforce: High turnover rates and challenges in managing a large delivery fleet.

- High Initial Investment: Significant capital required for app development and marketing.
- Limited Brand Awareness: As a new entrant, it lacks the recognition of established competitors.
- Logistics Challenges: Managing timely deliveries, especially in peak hours and remote areas.

Figure A. 1 SWOT matrices for Q1

Appendix B. BMC

Key Partners	Key Activities	Value Propositions	Customer Relationships	Customer Segments
Restaurants and Cafes: Collaborations to offer a wide variety of cuisines and exclusive deals. *Payment Processors:* Secure and efficient transaction handling. *Technology Providers:* Partnerships for app development, hosting, and maintenance.	*App Development and Maintenance:* Continuous improvement and feature updates for the app and website. *Marketing and Promotion:* Attracting and retaining customers and restaurant partners through targeted marketing campaigns. *Customer Service:* Providing exceptional support to resolve issues and enhance customer satisfaction	*For Customers:* Convenient and reliable food delivery from various restaurants. Real-time order tracking and estimated delivery times. Diverse dietary options. Exclusive deals and discounts. *For Restaurants:* Increased visibility and sales without needing their own delivery system. Access to customer Feedback and analytics. Marketing support through featured listings and promotions.	*Personal Assistance:* Dedicated customer service representatives for personalized support. *Automated Services:* In-app order tracking, automated notifications, and FAQ sections. *Community Engagement:* Loyalty programs, user reviews, and ratings to build a community. Social media engagement through contests, polls, and interactive content.	*Primary Customers:* Urban and suburban residents in Algeria who seek convenience in food delivery. Young professionals and students who prefer ordering food due to time constraints. *Secondary Customers:* Restaurants, cafes, and food trucks aiming to expand their delivery service reach without investing in their own delivery infrastructure. *Niche Customers:* Corporate offices and event organizers requiring catering services. Health-conscious individuals looking for specific dietary options (e.g., vegan, keto).
	Key Resources		**Channels**	
	Technology: Mobile app and website development, including backend infrastructure for seamless operation. *Human Resources:* Skilled personnel for app development, customer service, marketing, and logistics. *Partnerships:* Strategic alliances with restaurants, payment processors, and logistics companies.		*Mobile App:* Features like push notifications for order status updates and *promotions* *Website:* An alternative platform for browsing menus and placing orders *Social media:* Platforms like Facebook, Instagram, and Twitter for engaging with customers and promoting special offers.	

Cost Structure	Revenue Streams
Development Costs: Initial and ongoing expenses for app and website development. *Operational Costs:* Salaries for staff, customer service representatives, and delivery personnel. *Marketing Expenses:* Costs associated with advertising, promotions, and customer acquisition.	*Delivery Fees:* A fixed fee per order, which can vary based on distance and time. *Subscription Fees:* Premium memberships offering benefits like free delivery, priority support, and exclusive deals for different costumer category's *Advertising:* Restaurants can pay for featured listings, banner ads, and promotional campaigns within the app.

Table B. 1 Business model canvas

Appendix C. Sequence diagram

a. API Gateway

Acts as the entry point for all user requests. Routes requests to the appropriate microservice (Auth, Order, Delivery, etc.).

b. Service Communication

Microservices communicate with each other using HTTP/HTTPS or gRPC. Each service has its own database, accessed via the microservice.

c. Database Replication

Database updates are replicated to the replication server in Algeria. Ensure real-time or near-real-time replication to keep data consistent.

d. Security and Monitoring

Secure all communications with SSL/TLS. Monitor services using tools like Prometheus and Grafana, with logging handled by the ELK stack.

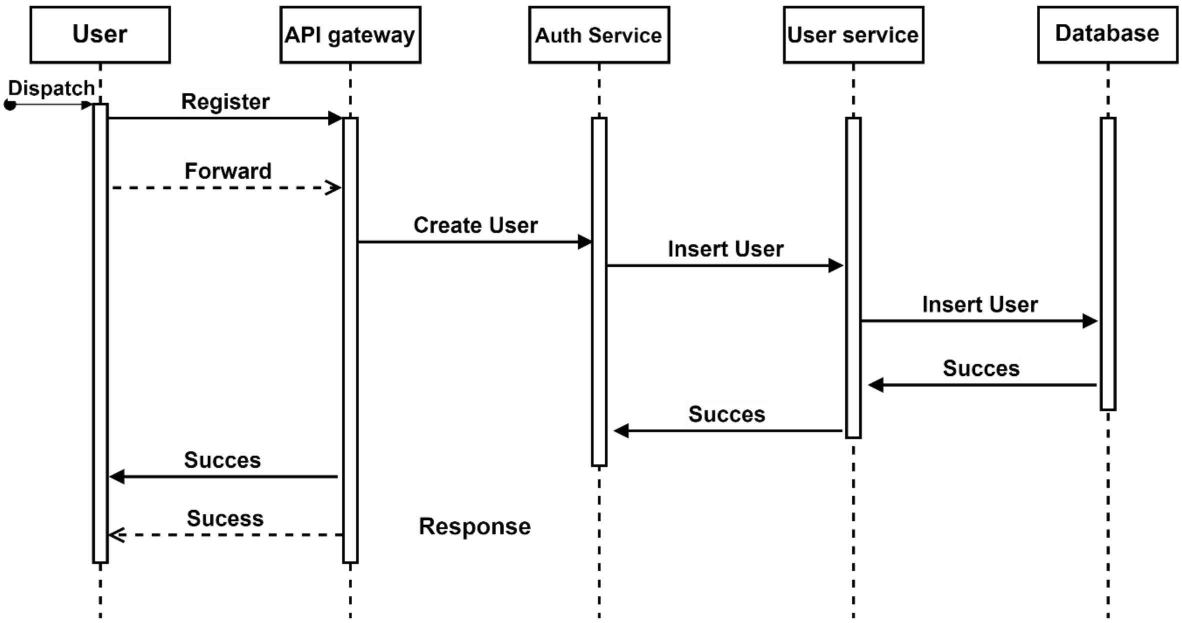


Figure C. 1 High-level sequence diagram of user registration

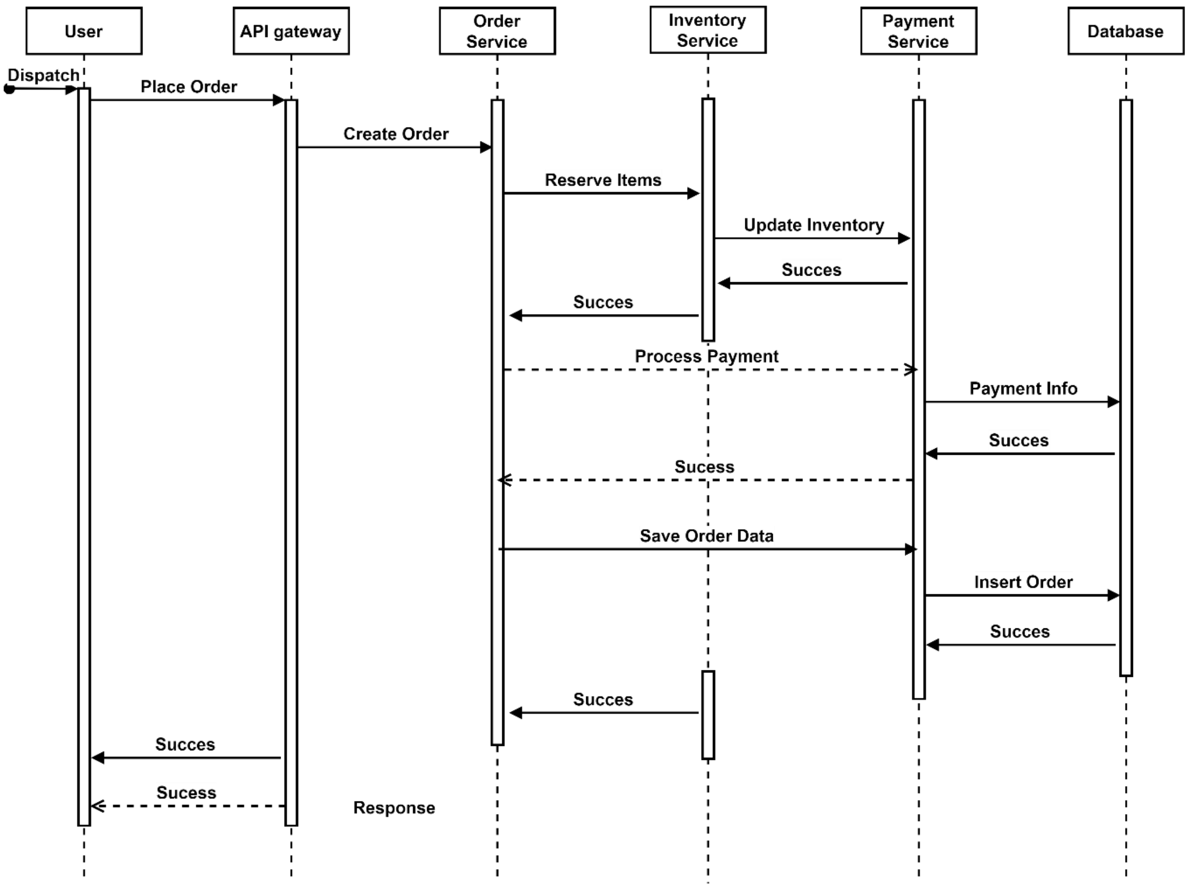


Figure C. 2 High-level sequence diagram of placing an order

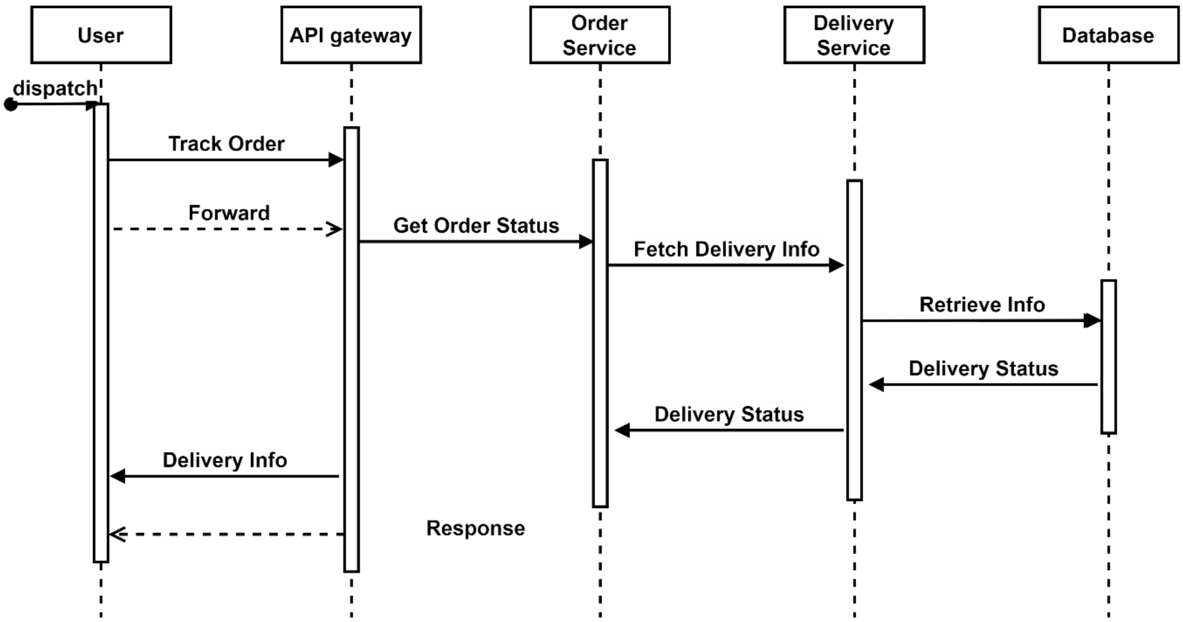


Figure C. 3 High-level sequence diagram of order delivery tracking

Appendix D. System design architecture

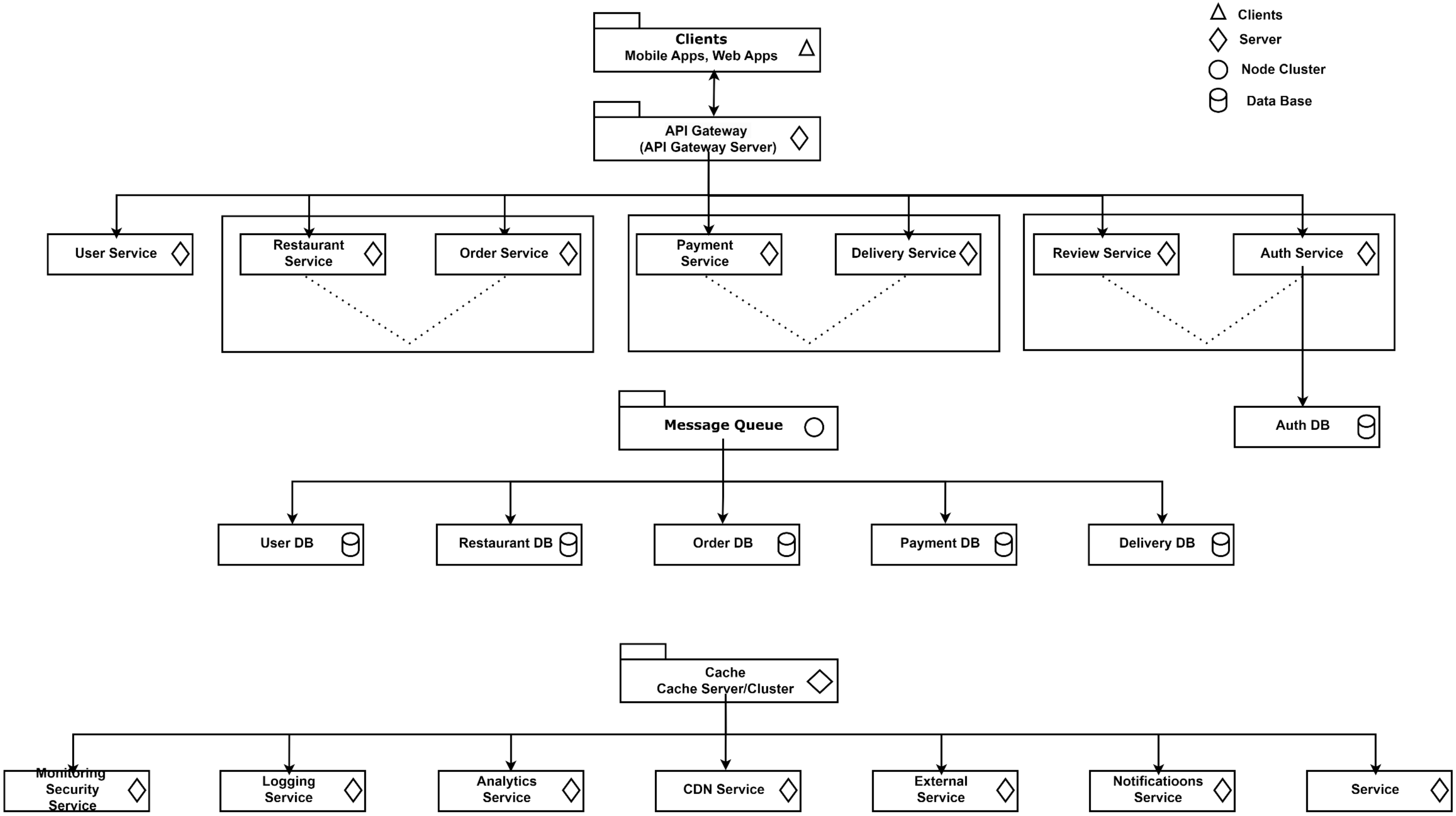


Figure D. 1 System design architecture

a. Primary Servers (Europe)

Server 1: API Gateway and Core Microservices

Server 2: Databases

Server 3: Auxiliary Services and Other Microservices

b. Replication Servers (Algeria)

Replication Server 1: Database Replication

Replication Server 2: Microservices Replication

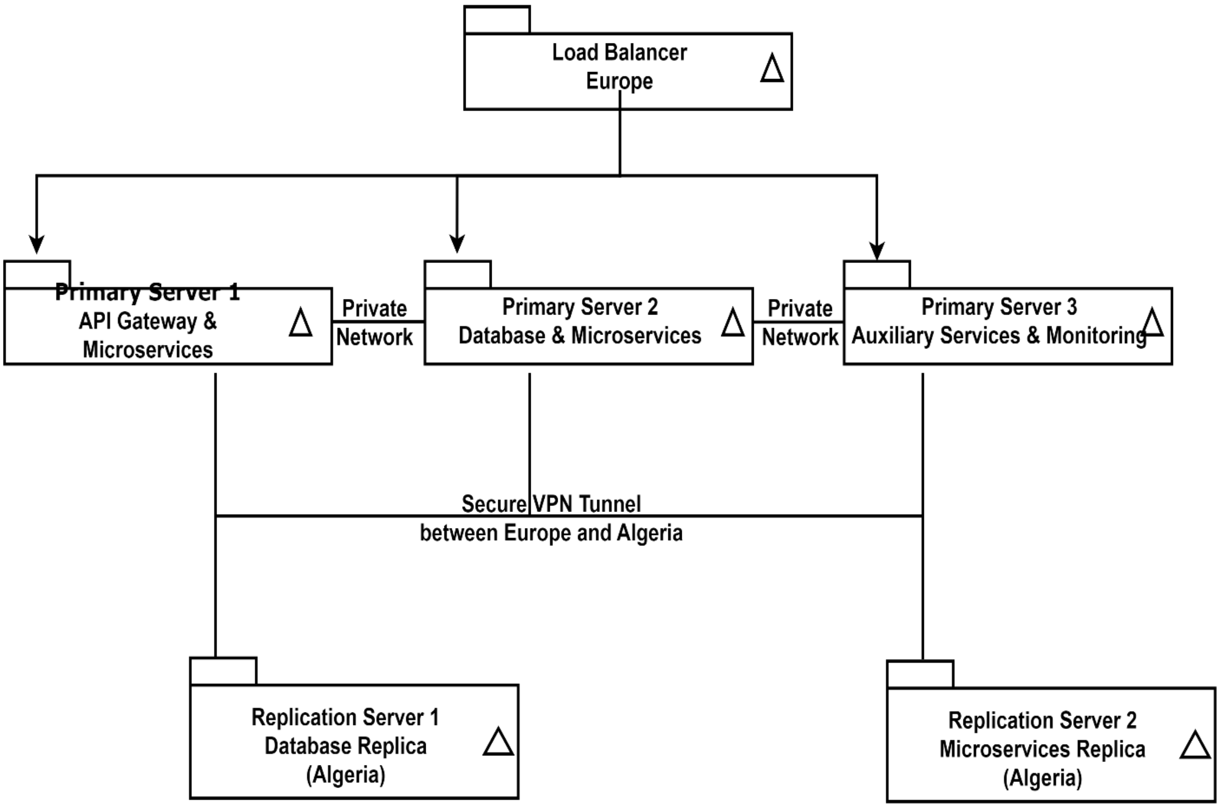


Figure D. 2 High-level server connectivity architecture

Appendix E. Certificate of domiciliation

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

جـــامعـــــة عمار ثليجي بالأغواط

حاضنة الاعمال الجامعية

حاضنة الأعمال بجامعة الأغواط
Incubateur Universitaire de Laghouat

رقم 14/الحاضنة/ 2024

شهادة توطين " مشروع مبتكر ضمن قرار1275"

أنا الممضي اسفله، السيد(ة): كراش شاكر عبد العزيز

مسير (ة) حاضنة الأعمال: حاضنة أعمال جامعة الأغواط

المقر الاجتماعي /العنوان: القطب الجامعي 2. جامعة عمار ثليجي بالأغواط

رقم علامة الحاضنة : 0911223047

تاريخ تسليم العلامة: 2022/11/09

أشهد أن الطالب / الطلبة التالية أسمائهم:

الاسم واللقب	تاريخ ومكان الازدياد	رقم تسجيل الطالب	الطور الدراسي	التخصص	الكلية
ATTATI MOUSSA	10/18/2001 Laghouat	191939002574	Master 2	Systèmes des Télécommunications	Sciences Technologiques
Mohammed AKROUT	07/29/1999 Laghouat	191939002808	Master 2	Systèmes des Télécommunications	Sciences Technologiques
SANDOUG Abdlazzize	01/29/2002 Laghouat	191939000617	Master 2	Systèmes des Télécommunications	Sciences Technologiques

قد قبل مشروعهم للاحتضان على مستوى حاضنة الاعمال لجامعة الأغواط تحت اسم:

Food Delivery SUFRAH

تحت اشراف الأستاذة **مسري مراد مخطارية** خلال السنة الجامعية 2024/2023.

سلمت هذه الشهادة بطلب من المعني(ة) للإدلاء بها في حدود ما يسمح به القانون.

حرر في الأغواط بتاريخ **2024/05/15**

مدير الحاضنة

مسؤول حاضنة الأعمال الجامعية
الجامعة الأغواط
كراش شاكر عبد العزيز