TRIBHUVAN UNIVERSITY

Office of the Dean

Faculty of Management

Kritipur, Kathmandu

An Internship Report

On

"COOP"

In partial fulfillment of requirement for the degree of

Bachelor of Information Management

(BIM)

Submitted By:

Laxmi Manandhar

Exam Roll No.: 7363/16

TU Regd. No. 7-2-551-104-2016

National College of Computer Studies (NCCS)

Paknajol, Kathmandu

April, 2021

NCCS Certificate

Organization Certificate

DECLARATION

I hereby declare that this internship project entitled "**COOP**" submitted to office of the dean, Faculty of Management, Tribhuvan University, is a result of my own internship study carried out in year 2020 at ITGlance Pvt. Ltd., Tripureshor, Kathmandu, for the partial fulfillment of requirement of the Bachelor of Information Management (BIM). It has not been previously submitted to any other university or anywhere else.

Laxmi Manandhar

TU Roll No.: 7363/16

TU Regd. No.: 7-2-551-104-2016

ACKNOWLEDGEMENT

I would like to sincerely thank Tribhuvan University, Faculty of Management and National College of Computer Studies (NCCS) for providing the opportunity to carry out this internship project that helps to enhance our knowledge, skills and capabilities.

I would like to thank ITGlance Pvt.Ltd, which accepted my presence and provided me to work in their organization. It was a great experience to practically learn and closely observe the organization environment and work culture. I had the greatest opportunity to share views and ideas with different people in the organization environment and work culture during my internship period.

I would like to take this opportunity and show my gratitude towards Mr. Ganesh Man Singh (Principal, NCCS) and Mr. Santosh Maskey (Vice-Principal, NCCS), Mr. Prajwal Baniya Chhetri (Program Director, NCCS), Mr. Rajan Poudel (Program Coordinator, NCCS) and Mr. Umesh Maharjan (Supervisor, NCCS) for providing me the necessary guidance and supervision.

I would like to express my greatest gratitude towards Mr. Yogendra Maharjan (CEO, ITGlance Pvt. Ltd.), Mr. Prabin Byanjankar (PM, ITGlance Pvt. Ltd.) and Mr. Roshan Limbu (Mentor, ITGlance Pvt. Ltd.) for providing direction in the intern process, guidance, suggestions and timely help during the internship period. Finally, I am very much thankful towards all the teachers, teammates and friends who supported me as well as provided all the facilities for the successful completion of this report.

Laxmi Manandhar

ABSTRACT

COOP is established to carry out various transactions and monitor the financial transactions and accounts. The system enables individual customers, business owners, corporate companies and institutions to make easy transactions and working processes in one platform. This software's main goal is to make cooperative work easier and faster. Customers can easily deposit and withdraw money, take loans etc. The document begins with the background of the related field along with the objective for preparing the report. It also states the methodology, tools used in the preparation of this report and the development of the software system.

The document consists of the brief introduction of the information technology and software industry. Also covering the information of the framework and project management tools used in COOP.

The brief introduction of the selected organization is highlighted with the objectives, vision and mission of the organization. The structure of the organization has also been mentioned to display its working process.

Introduction to evolution of IT along with the objective of making the world a better and easier place is highlighted. More emphasis is given to the Nepali sector, how IT has helped in Nepal and yet, the limitations and challenges too.

The software is a digital solution for financial institutions that stores the client's information, their associated products such as loan, deposits, withdrawal, etc. with necessary transactions that occurs in the institution. The software's main goal is to make cooperative work easier and faster. Also, the analysis of the activities carried out during the internship program. It consists of the information of the division of modules comprising the COOP system. The details of modules worked on are included here. Along with this, system analysis, design and testing are also included in this section.

Finally, the lesson learned both managerial and technical ways and is defined in the summary of the documentation.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Programming Interface
BIM	Bachelor of Information Management
COOP	Cooperatives
HTTP	Hypertext Transfer Protocol
IT	Internet Technology
JWT	JSON Web Tokens
KPI	Key Performance Indicator
NCCS	National College of Computer Studies
NTA	Nepal Telecommunications Authority
OOM	Object Oriented Model
PM	Project Manager
QA	Quality Assurance
SDEs	Software Development Engineers
SDLC	Software Development Lifecycle
TIBCO	The Information Bus Company
TU	Tribhuvan University
USP	Unique Selling Point

Chapter 1

INTRODUCTION

1.1 Background

Bachelor of Information Management (BIM) is a four-year course offered by TU that provides a unique blend of 60% information technology and 40% management courses which helps students to gain the technical expertise and various management concepts required in an organization. It is conducted by major Universities in Nepal in the semester system which is divided into eight semesters and 126 credit hours. This temporary job program has been planned for the halfway satisfaction of the level of BIM.

The objectives of the BIM are as follows:

- Production of the professionals who have management skills as well as the knowledge on Information Technology.
- Prepare IT professionals proficient in the use of computers and computational techniques in order to develop effective information systems to solve real life problems in the organizational environment.
- Production of IT professionals who are skilled at the computers and computing techniques in order to create effective information system which could solve the organization's problems.
- Develop ability in students in object-oriented software design methods and data management systems.

Internship is the situation of an understudy or student who works in an association here and there with or without pay so as to pick up work involvement or fulfill necessities for a capability. It is characterized as an understanding between an understudy and an association where an assistant fills in as a representative for the association for certain lime span. It is an impermanent position with an accentuation on hands on preparing instead of only business. It gives a stage to be engaged with a genuine working situation and create required vital aptitudes. The internship is one

of the programs designed by the faculty of management, Tribhuvan University. It is a pre-professional work experience that provides students with the chance of gaining experience in a career field. This program helps us to know how the organization as a whole operates and also how communication takes place and activities are coordinated in the real world. Thus, the internship program can be regarded as a fruitful opportunity for the students of BIM to be familiar with the actual work environment of Nepal. An internship can give us a real insight into the world of work, allowing us to build on the theory we learned at university and helping us to gain practical skills that will help strengthen our CV and make us more employable. Internships also offer the chance to test our skills in real-life situations, explore our career options and gain an insight into an organization or career path.

This report has been prepared based on the internship experience of 3 months in ITGlance Pvt. Ltd. It has provided real life experience and knowledge of how actually is the work place and work environment. The theoretical knowledge that I gained in the college was applied in the actual field specifically in programming. This internship has not only sharpened my skills in programming, but also increased my self-confidence, self-motivation and self-development.

1.2 Focus of the Study

Internship is one of the most vital part in the student's life, who are about to start their career. It provides real life experience about the work, procedures and methodologies followed in the organization. This study focuses on the following areas:

- Career opportunities

  While working in an organization, students can get different opportunities to start their career. The knowledge they gain there can be very helpful in the future to work as an employee.

- Exposure

  The students can get exposure while working in an organization. They get familiar with the working procedure and can be able to work under pressure. Also, they get in touch with various people who can help them in the future.

- Gain real life experience

  One of the main purpose of internship program is to allow the students to gain real life experience. Students are restricted inside the classroom and do not get any platform to sharpen their skills. So internship provides a great platform where students can learn and showcase their knowledge and skills.

1.3 Statement of Problem

The main problem seen was no other digital way of having the transactions recorded and used.

- Manual record of information in small cooperatives

  All transactions were being done manually. The recordkeeping was main hassle faced by teller and managers.

- Loss of hardcopies and data entry errors

  Manual recording made it difficult for managing the files, saving them from damage and loss over time. It was also so difficult for them to search for entries, let alone the errors caused while manually recording the data.

- Perplexed record keeping and decision making for the managers

  Since this is a financial system, different KPIs are necessary for deciding what to keep, how to balance sheets, where the amounts have been collected from, etc. Large sets of data and financial transactions were to be manipulated but was being difficult for managers to do so.

1.4 Objectives

1.4.1 Objectives of Internship

The most important objective of an internship is to make students familiar with the real working environment. By doing an internship, we get all of the personal and cultural perks while simultaneously reaping the professional benefits. The objectives of the internship program are as follows:

- To build interpersonal skills and confidence
- To implement the theoretical knowledge acquired from college in real working environment
- To pet career insight existing in the country
- To receive an introduction to an organization within the Information systems industry and to gain a better understanding of its design and structure.
- To get familiar with the real working scenarios, and accordingly tackle them

1.4.2 Objectives of System

The objectives of the system can be listed as follows:

- To digitize financial operations
- To provide figurative and interactive graphical interface for decision making
- To effectively manage cash flows and lessen data irregularities

1.5 Methodology

During internship program, I have collected the information about the cooperative management system. The information was collected with the help of primary and secondary sources. There was much information that is collected directly from the organization through day-to-day operations. Also, the information gathered from the experience is included in this section. Much information was gathered from websites, manuals and other sources of the organization. Secondary data or the references are readily available data which are inexpensive to obtain. These types of

data are administrative data which are collected over a long period by the organization themselves.

1.5.1 Organization Selection

For the internship to be worthwhile, selection of organization is an essential factor. it is important to complete an entry level position. It is a medium through which we will pick up understanding and presentation to the outside world. The organization that we choose plays as the medium through which we get exposure to real world. The organization I selected is ITGlance Pvt. Ltd. which was established in 2008. As per my interest, I had to choose a software company to enhance my skills and knowledge in the practical field. ITGlance is a systematized software company in Nepal, from where I gained valuable experiences and knowledge.

1.5.2 Placement and Duration

Organization's selection process placed me as an intern in the organization where l got the opportunity to learn the real organization process and software development process along with the importance of teamwork in achieving success. I was trained with the organization's work process that lead to an easy path for organization culture and process.

Table 1-1 Internship schedule

Start Date	August 1, 2020
End Date	December 1, 2020
Days of Work	Sunday to Friday (6 Days)
Office Hours (Work from home)	9:00 AM to 6:00 PM (9 hours)
Position	UI/UX Designer
Internship Period	3 months

1.5.3 Work Procedure

For the initial week, I was introduced to the working environment of the company to make me feel comfortable. The second week was a mere orientation on designing concepts and design thinking. Third week went by with designing simple portfolio sites. I was given knowledge about the basics of user experiences, persuasion and design psychology that are essential for carrying out the project more efficiently in the fourth week. I also learnt to operate various application for carrying out my project smoothly which will not only help for the successful accomplishment of my project but will also provide guidance while doing some other projects. After 4 weeks of learning about the basics, I was given a design project that is being developed under java i.e. COOP. We were working from home due to lockdown situation. Yet, managed to have daily standups, meetings, project requirements collection and worked carried out in timely manner, just how we would physically in the office hours.

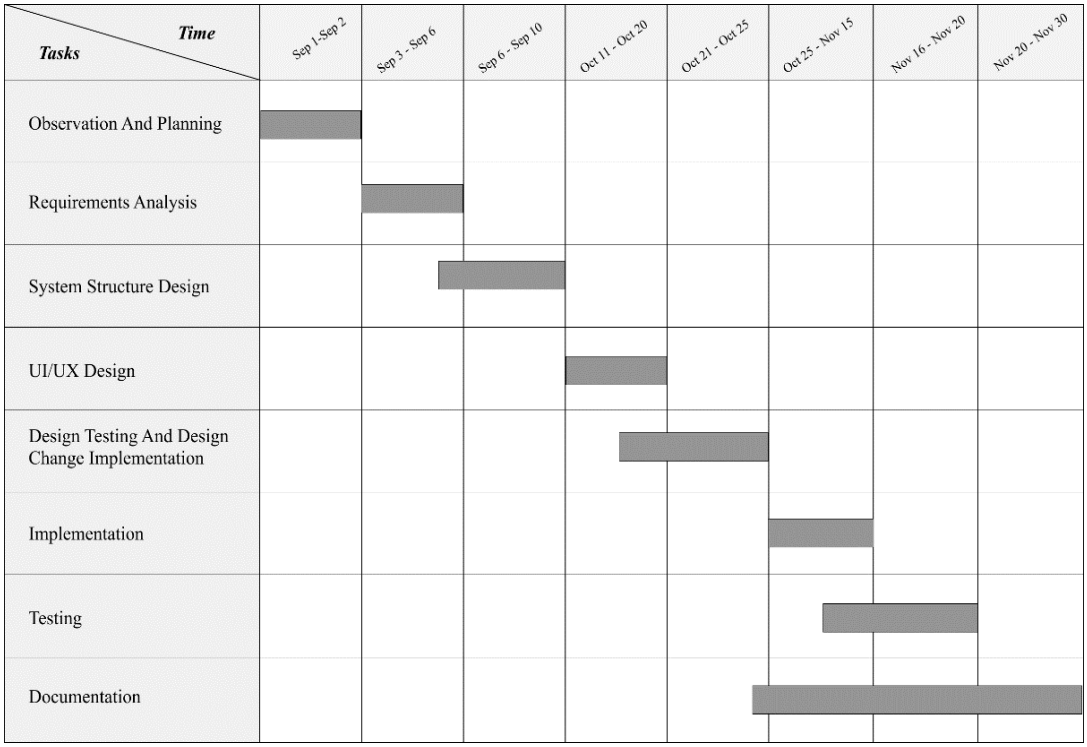


Figure 1-1 Gantt chart for COOP software development

Requirements were gathered and planned carefully. Then those requirements were analyzed, whether they were feasible or not, how the system cycle will proceed on. The requirements were studied and a basic overview of the system was designed with interaction of the project manager, supervisor, developer team and design team. According to the system structure designs such as classes and other backend connections, the mockups were created. Mockups were connected and prototyped. Not just the design was to be made but also to be tested. Whether the client approved the designs or not, how well the users were accustomed to it, etc. were tested. The implementation of design took place and also the implementation. Again, the developed product was tested. The entire time, documentation was being carried out for the stages so that it would be easier to communicate with client, project manager and developers.

1.5.4 Tools Used

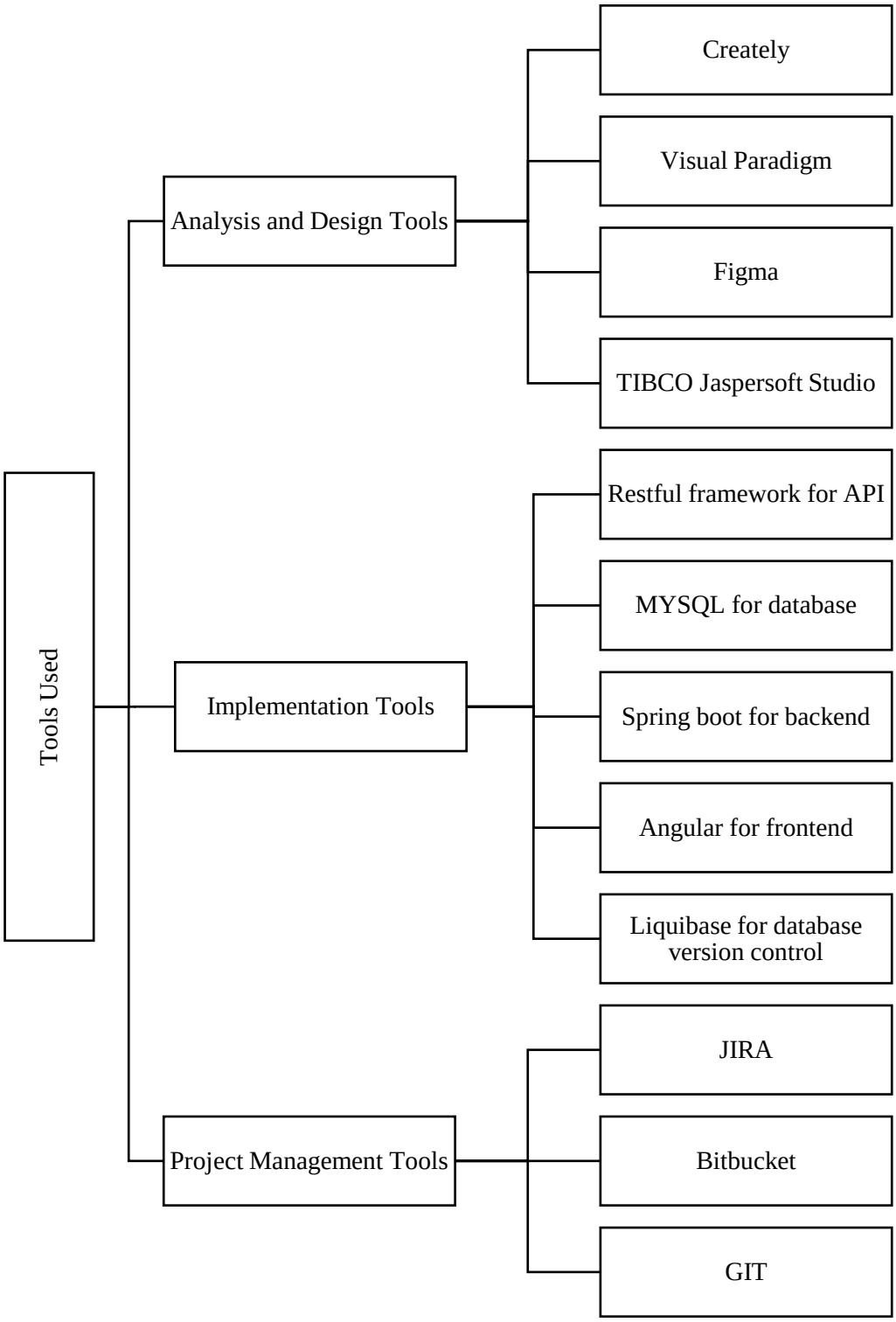


Figure 1-2 Tools used

1.6 Limitations of the Study

It was a great opportunity to be an intern in the IT Company. This report is prepared based on the observations, experience of the internship formal and informal interview with the staffs and the secondary data available in internet. However, the internship had some limitations. The main limitations of internship are listed below:

- Due to lack of time constraint, study of each and every activities of the IT Company in depth was not possible.
- Due to security policy research was done in surface and not in depth.
- Due to privacy policy maintained in an organization there was unavailability of information.

Chapter 2

INTRODUCTION OF INDUSTRY

2.1 Introduction to Information Technology

Information technology is one of the growing sectors in Nepal which is just on the starting phase but has lots of potential for growth, investment and significant profits. Software companies are growing like mushrooms as users buy computers for home and business use and need something to do with them. Several industries are associated with information technology, including computer hardware, software, electronics, semiconductors, internet, telecom equipment, e-commerce and computer services. Both software development and the hardware involved in the IT industry include everything from computer systems, to the design, implementation, study and development of IT and management systems. Owing to its easy accessibility and the wide range of IT products available, the demand for IT services has increased substantially over the years. The IT sector has emerged as a major global source of both growth and employment.

The government of Nepal has also identified IT as one of the five priority potential export service sectors in Nepal Trade Integration Strategy (NTIS) 2010. The IT industry has become an indispensable part of our economy as it helps in creation of more knowledge-based economy, human resource development, assist in effective management of other services and creates various other opportunities that aids in development of a nation. Using information technology. individuals and businesses have the ability to view changes in global markets at real time far faster than they usually used to do.

2.2 Introduction to Software Industry

As we know that the world is moving from analog to digital, PT is playing an important role in the people's life. IT has become a part of life for humans. As the world is becoming narrower and narrower, the influence of TT is becoming more

and more. There are 17 revolutions in all of the countries in the world, Due to the help of IT people, generating new techniques to improve their skills and talents.

Nowadays, in all operations the involvement of information technology is vital. Information Technology helps in building the nation's strength, and also provides a way to economic development. Today, information technology continues to advance and change the way of business work in unexpected new ways. IT can be defined as a contemporary term that describes the combination of computer technologies with the telecommunication technologies. Its vast power to affect change has put it in the forefront of all other industries. Today, IT is the fastest growing economic activity in the world and IT industry as such is the biggest industry.

IT has helped computerize the business process thus streamlining businesses to make them extremely cost-effective money-making machines. Thus, in turn increases productivity which ultimately gives rise to profit that means better pay and less strenuous working condition. IT has made it possible for businesses to be open 24 x7 for business over the globe.

The IT industry can serve as a medium of e-governance, as it assures easy accessibility to information. The use of information technology in the service sector improves operational efficiency and adds to transparency. It also serves as a medium of skill formation. In this new era, IT stands as the central force in shaping organization, societies and nations based on the presumption that it is the key to achieve progress today. At last, Information technologies are the product of developed countries and to make that technology suitable for developing countries like Nepal. There should be an effort to build a capacity to recognize the importance of implementing IT according to the development needs and requirements. So, IT is the most vital component to build up the nation and helps the economic growth.

2.3 Objectives of Information Technology in Business

- To ensure the availability of and access to information that enables customers to make timely, informed decisions by strengthening data and knowledge management approaches, ("Department IT Strategic Goals", 2010).

- To explore new horizon of challenges and opportunities that can boost the technological, economic and educational scenario of the country
- To manage the cost efficiencies
- To provide a robust and secure IT infrastructure that supports on-demand access to information
- Globalization
- Promoting the growth of e-commerce and software export
- To improve end-to-end transparency and accountability between the customer and service provider
- To increase percentage of employees who enroll in IT training opportunities

2.4 History of IT in Nepal

In this revolution of IT, we have witnessed many changes from machine era to automated era. Coming to this long we are now totally living in a world of technology. A brief history of IT in Nepal is shown below:

- Use of computer for the first time in history of NEPAL in 1971 census by NCC.
- Another 2nd generation computer was used in 1981 census named ICL 2950/10.
- Telephone Service exchange was established by NTC in 1960 in Kathmandu Valley.
- RONAST E-Mail Service.
- Mercantile Communications in 1995 started Internet Service for the 1st time.
- In 1998, NTA, a telecommunications regulatory body, was formed as per the Telecommunications Act 1997, (Dhungel, 2019).
- Involvement of ISPs
- IT policy 2000 - place Nepal on global IT map, to make IT accessible to general public, creation of knowledge-based society/industries, (Bhurtyal, 2010)

- Telecommunication Policy 2004 - to create favorable environment to make the telecommunication service reliable and accessible
- eGovernment Master Plan 2005
- Electronic Transaction act 2008 - ADB supported ICT Development Project

Due to success of NCC's, there are more than 300 legal private computer training institutes. I.T Ministry established for advising and formulating plans and policies by NIC instead of NCC. NTC provide cellular and WLL network to private parties UTL, NCELL, Hello Nepal. NTA provide license to various ISPs like- Worldlink Communications Private Limited Broadlink Network and Communications Private Limited Hotlink Nepal Private Limited. Traditional method of business and education are now replaced by advance I.T innovations, (Soft, 2019). Rapid and efficient development of Telecommunication and transmission media. Now there are more than 4 million users of internet. Although there are a lot of encouraging signs for Nepal in the field of IT, the current quality of IT services and education still leaves a lot to be desired. Considerable amount of work needs to be done and the initiative should be taken early if we are to level and keep up with the world in the field of technologies. 21st century demands globalization.

2.5 Opportunities in Nepali IT Sector

Smart irrigation project, digitization of land records, e-Haat Bazar are some of the services that ICT can contribute to the development of agriculture. Smart classrooms, online learning platforms, biometric attendance systems, and mobile learning centers in rural areas can bring massive reform in our traditional education system. Electronic visas augmented, and virtual reality tours, electronic tour guides, tourist tracking systems, online information regarding tourism and websites, and mobile apps can expand the scope of tourism globally. Urban infrastructures such as smart building, intelligent waste management, public transport management, municipality mobile application, etc. could lead to a creative and prosperous Nepal. Electronic health records, mobile health units, centralized telemedicine centers

would help in gaining better health care. Smart grid projects, smart metering can improve our energy system go digital. National payment gateway, information management system, mobile wallet system can lead to financial reform. ICT plays a significant role in disaster prevention, mitigation response, and recovery. The use of Drone for security monitoring and delivery of emergency equipment can help in easing disaster. ICT has a consistently positive impact on job creation. The opportunities exist in infrastructure development, equipment manufacturing, distribution, maintenance, and also in the event of websites, mobile apps, information systems, etc. (Aryal, 2020)

2.6 Challenges in Nepali IT Sector

The successful use of ICT in these various sectors will require a high degree of emphasis on implementation. The Government of Nepal needs to focus on the following priority areas to create an enabling environment: Technology and Infrastructure development, entrepreneurship/PPP so that private organizations would involve in every sector, talent, and skills development so that country will get the skilled technical workforce to implement the efficient use of ICT. Encourage private sector participation, improve professional education, facilitate the event of a robust financial ecosystem, encourage foreign direct investment in priority areas, making public servants digitally-ready will be essential. Emerging business models and disruptive technologies such as artificial intelligence (AI), robotics, the Internet of things (IoT) and over the top (OTT) are transforming the way work is done. These emerging technologies are allowing governments and enterprises globally to unlock the potential to achieve exponential growth. (Aryal, 2020)

Chapter 3

INTRODUCTION OF ORGANIZATION

3.1 Introduction of Organization

ITGlance Pvt. Ltd. is a professional Software Company in Tripureshor, Kathmandu, Nepal. They are the IT organization who give the best benefits and custom-made frameworks as per the prerequisites of the customer. ITGlance pursues the dexterous technique to isolate the errands into littler dashes of one week. They take a shot at the premise of customary criticism with the client, executing new advances for a customer to meet their prerequisites effectively and bringing people together as efficient, highly-collaborative teams to continuously deliver value to users.

They believe in providing cost effective and reliable services to their clients with their highly skilled team members. Their expert team is always ready to serve you with professional custom website design, web development, e-commerce website design and development, community websites design, website redesign etc. and also the maintenance service for your website. Their experts are well skilled and can build stylish as well as elegant and easy navigated websites for you. Over the years of experience, ITGlance Pvt. Ltd. has achieved a prominent position of an expert software company possessing some of the best analytical brains. Their transparent, efficient and flexible world class software development process zero downs risks of project failures and creates powerful software solutions that meet present as well as future demands.

3.2 Vision and Scope

Customer loyalty through customer satisfaction, delivery in time, quality assurance and management, innovation to improve efficiency, live support and to be recognized as the best company when it comes to delivering value to customers.

3.3 Services Offered

Primary services offered by ITGlance are:

- UI/UX and Graphics Designing

ITGlance has been offering creative UI/UX and graphic design solutions to a variety of business needs. It offers a complete array of graphic services that combine strategy. Creative thinking and technical expertise in making designs stand out and deliver. It also provides absolute solution to the businesses or individuals, to post simple information about themselves or about their company onto the web page. They have been providing a complete dynamic web solution as per the client's expectations. It designs and develops dynamic web sites as per specific requirements of clients which are flexible, user friendly, with effectiveness and a great amount of interactivity. The prototyping service really makes it easier for both clients and developers.

- Software Development

ITGlance has been providing custom software using various programming languages like C, C#, java, react and .NET. They deliver rapid application development processes and thoroughly tested and completely documented systems that can be deployed on any major IT platform. Whether you want a single or multi-user desktop application, or an enterprise-wide server-based solution, ITGlance have the knowledge and experience to get the job done.

- Online Promotion

When correctly executed, online marketing can be one of the most cost-effective and efficient ways of reaching your target audience. The online marketing services include: search engine optimization (SEO), pay-per-click (PPC) advertising, email marketing, social media marketing, Google Analytics, Google Website Optimizer and training.

- Search Engine Optimization

Search engine optimization is a methodology of strategies, techniques and tactics used to increase the number of visitors to a website by obtaining a high-ranking placement in the search results page of a search engine including Google. Bing. Yahoo and other search engines. ITGlance implements ethical and effective

search engine optimization strategies for websites so that it can enjoy top page ranking in Google and other major search engines naturally building on online presence for the site through social media as well.

- Web Hosting

Web hosting is the activity or business of providing storage spaces and access for websites. ITGlance servers are Linux-based. high-end Web servers designed from the ground up with small business requirements in mind. The combination of the Linux server platform and the state-of-the-art data center translates into the most stable environment available for your small business website. email accounts and databases.

Other services offered by ITGlance are:

- E-Commerce Application Development
- Integrated Payment System
- Content Management Solutions
- Web Portal Development
- Flash Developments
- Graphics Designing
- CSS
- Hardware Support and Maintenance
- Network Management and Maintenance

3.4 Organization Structure

Organization structure is a system that is used to define a hierarchy within an organization. It helps for running the organization in a smooth manner. Therefore, ITGlance has a systematic structure consisting of CEO, Technical Director, Developer, Senior Programmer and Junior Programmer who handle different field of the organization.

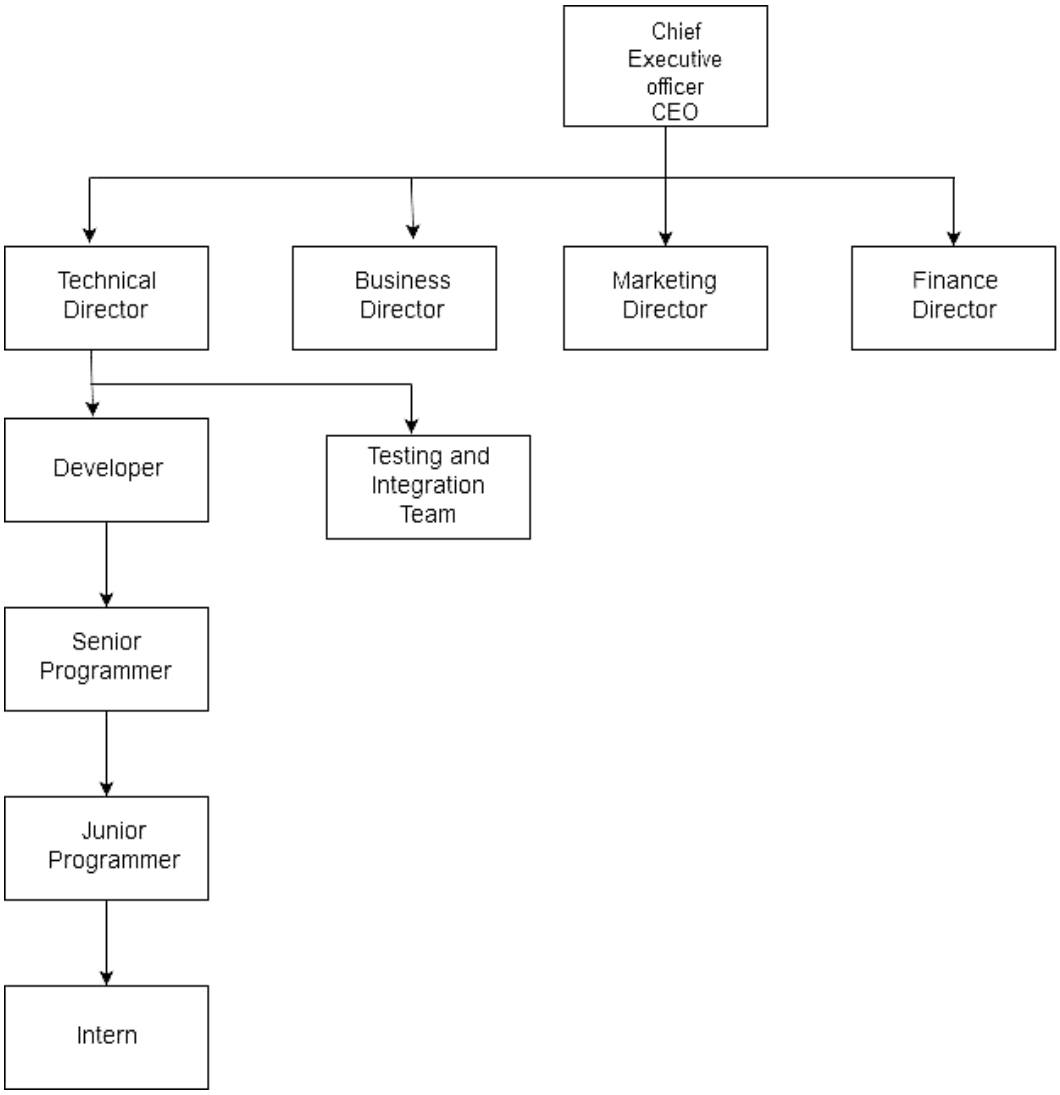


Figure 3-1 ITGlance organization structure

Chapter 4
SYSTEM ANALYSIS, DESIGN AND DEVELOPMENT

4.1 Introduction to the System

COOP is an application that coordinates and integrates all the activities and transactions involved in running and working of cooperative organization. This software is developed in order to simplify the transaction and working process of cooperation. This system provides users with the facility of performing every transaction of work involved in a cooperative system. The main idea behind developing this project is to simplify the transaction including the attributes such as credit, debit, deposit, client, transaction, employees, interests, loan etc. making it secure and simple for users to work using IT technology. This organizational application not only helps to ensure the easy working of entire organizational work but also helps to ensure the secure and reliable work done.

The employees of cooperative organizations can use the application so as to perform the related banking task avoiding all the paperwork and helping employees to have their work done in minimal time as possible. The system not only allows users to do their input and their work-related tasks but also keep the record of all the transactions in a very secured database. This system can also be used for calculating interests to their loans provided. It takes all the records of accounts associated with their respective field of attributes. This application keeps the records of the related account work such as loans, interests and required calculation needed.

4.2 Objectives of the System

- To digitize financial operations
- To provide figurative graphical interface for decision making
- To effectively manage cash flows and lessen data irregularities

4.3 Functions of the System

- Loan Management
- Deposit Management
- Share Management
- Client Management
- Employee Management
- Report Generation

4.4 Requirements of the System

4.4.1 Functional Requirements

These are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations. COOP should be able to perform these activities:

- Cashier should be able to deposit payment, make transfer and withdrawal
- Loan officers should be able to design load products, set criteria and approve loans
- Manager should be able to design products
- Each user of the system should be able to login and logout from the system
- Each user of the system should be able to update his or her profile change password, etc.
- Accountants should be able to manage debit and credit and generate financial reports
- Clerk should be able to register new customers and maintain customer records
- The system should be able to generate loan repayment schedule automatically

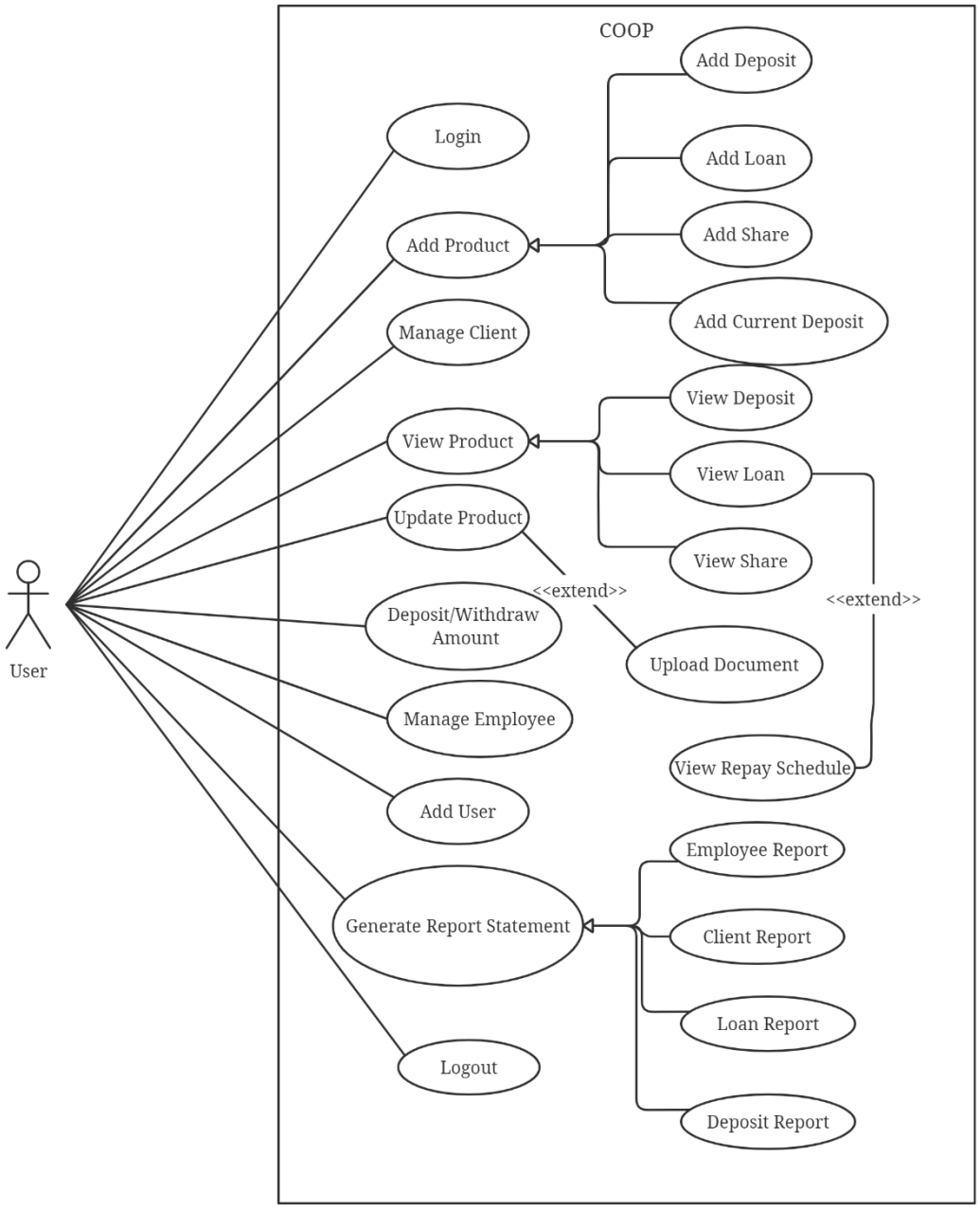


Figure 4-1 Use case diagram

4.4.2 Non-Functional Requirements

Non-functional requirements are requirements that are not directly concerned with the specific delivery by the system to its users. They may relate to emergent system properties such as reliability, response time and store occupancy. NFR may come

from required characteristics of the software (product requirements), or from external sources, ("Non-functional Requirements", 2021).

- **Product requirements**

  Memory

  The system must be taking up less memory space for data manipulation.

  Accuracy

  The results must be accurately displayed according to the query, meaning that all the programs written must be tested properly with multiple possible inputs.

  Speed

  Reduce unnecessary dependencies, limit the use of packages and create manageable modules for each operation for smooth operation.

- **External requirements**

  The system must comply with legal and financial acts. It should maintain transparency but should not disclose information to unrelated parties. The system should be liable for providing ease of access for the users.

4.5 Feasibility Analysis

The main objective of feasibility study is to test the technical, operational, economic and schedule feasibility. All systems are feasible only if they are given unlimited resources and infinite time. It helps to determine the benefits of the proposed system in the society and organization. It also determines if the system can be built successfully with cost, time and effort, ("Technical Writing", 2021).

4.5.1 Economic Feasibility

It contains the most basic features about financial operations like daily transactions, report generation, products management. The technologies and resources needed to

build the software, is already available. users only need an internet facility to access this software. So, this software is economical and can serve the user's purpose.

4.5.2 Operational Feasibility

There are some challenges such as users being reluctant to change, but there is high operational feasibility in this system otherwise, because of simple user interface for the users. Also, all the features will be implemented using its own databases and through API and it is compatible for all devices. Therefore, this system will meet the organization’s operational requirements.

4.5.3 Technical Feasibility

The technical feasibility assessment is focused on gaining an understanding of the present technical resources available and their applicability to the expected need of the proposed system. All the necessary technology such as Spring Boot, MySQL, etc. are already available. Also, other resources like laptop, internet, etc. are available. This is a web-based application so it can run on all platforms. There will not be any problem while running this software on any personal computer, making it technically feasible.

4.5.4 Schedule Feasibility

As for the features that we have planned and analyzed for this phase and release, the system has high schedule feasibility. For future works too, there are more things to be explored, such as client requirement changes, which can also be amended in no time. Hence, we do save both time and cost with this system.

4.6 System Design and Modeling

4.6.1 Data Modeling

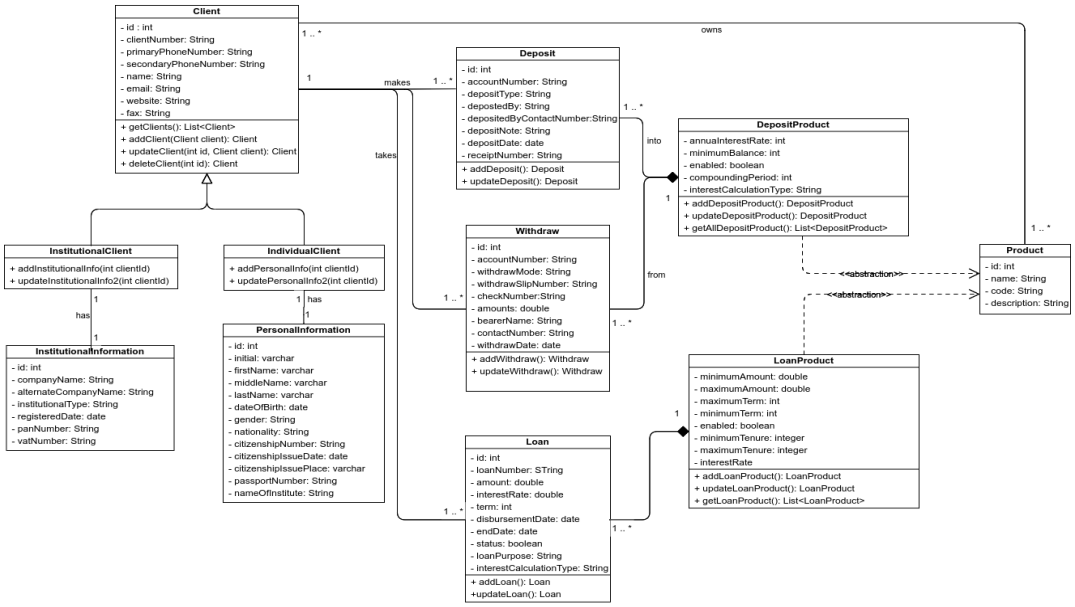


Figure 4-2 Class diagram

Above is the class diagram of the system. A class diagram is a type of diagram and part of a unified modeling language (UML) that defines and provides the overview and structure of a system in terms of classes, attributes and methods, and the relationships between different classes (technopedia, 2021).

The main classes associated with the system are Client, Product, DepositProduct, LoanProduct, Loan, Deposit, Withdraw, Employee, etc. As described in the figure, the client can be of two types i.e. Institutional Client or Individual Client. Institutional Clients can have an extra attribute called institutional information containing organizational details and the Individual Client can contain the personal information. The Product class defines the services that the co-operative has to offer. It could be a deposit product, loan product or share product. On the deposit product, transactions like deposit and withdraw can be held which are defined by the Deposit and the Withdraw classes respectively. Likewise, disbursed loan information is defined by the Loan class.

4.6.2 Process Modeling

In the given figure, the activity diagram of the system is illustrated. The activity diagram defines the actual flow of the system. Firstly, the user needs to login to the system. If the provided credentials are correct then only the flow continues, else the user is prompted to the login page again. After successful login to the system, a dashboard is displayed where various KPIs are included. Now, the user has liberty to perform various operations such as exploring client section, product section, teller section, organization section, report section and lastly configuration section. If the user does choose the client section then the flow of the activity becomes: Search the client according to the client name or number then view certain client's detailed information then view the client's associated products and lastly to add new associated products.

Similarly, if the user chooses the product section, then the user can search for the client with their name or number for their related product i.e. loan and deposit. If the user chooses the teller section then the user can make deposits and withdrawal from the respective accounts. For the organizational section, the user can search the employees, add the employees and view the details of the employees. For, report section, the user can generate employee, client, loan disbursement and deposit reports. For the configuration section, users can configure the users as well as configure the products. Lastly, the activity flow ends.

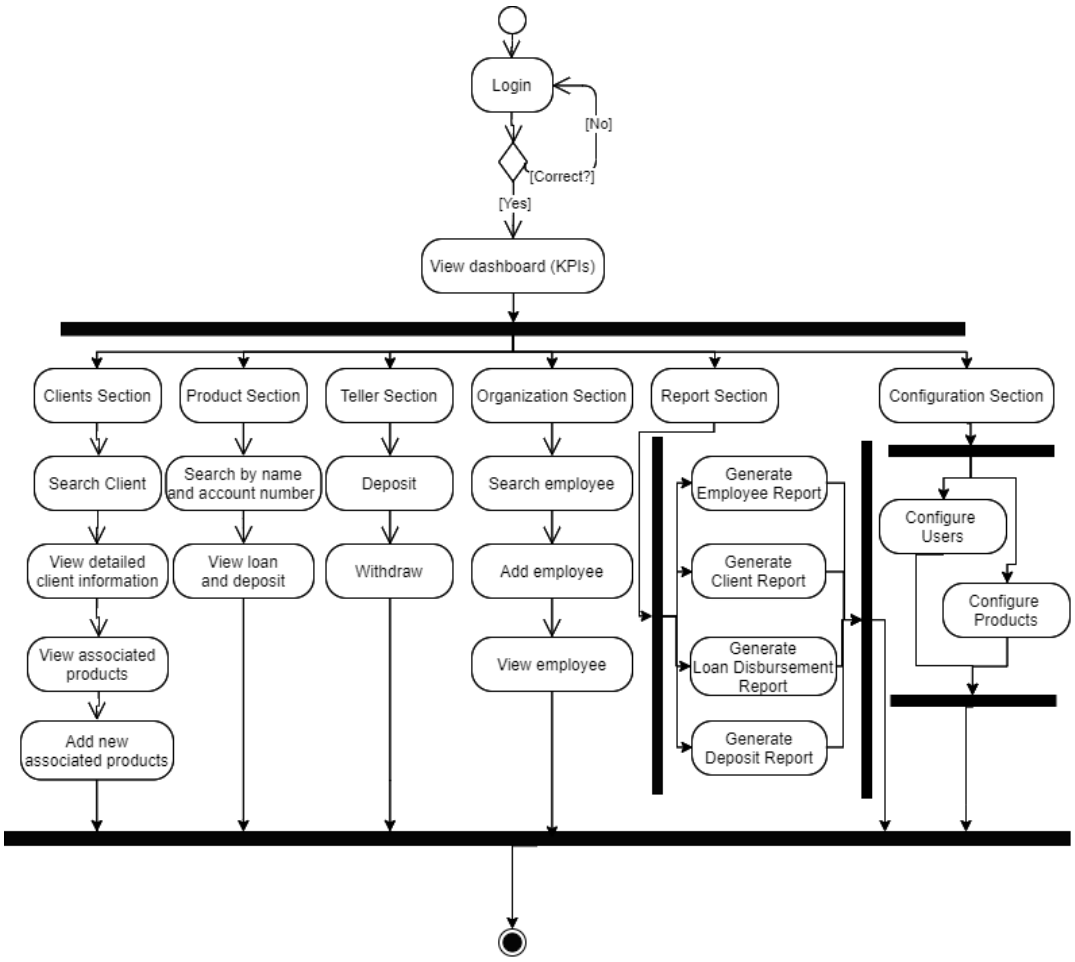


Figure 4-3 Activity diagram

4.6.3 System Architecture

System architecture is an overview of the system in response to the conceptual and practical difficulties of the description and the design of complex systems. System architecture is a conceptual model that defines the structure, behavior, and more views of a system, ("Software Architecture", 2021). It involves a block diagram which is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that shows the relationships of the blocks. Here, COOP provides different services to the user of the system i.e. employees of the cooperative banks which can be operated through a web browser using the internet. Such services are recording client's details, recording every day

transactions, making deposits, withdrawing, issuing loans, etc. A basic overview for employee is shown below:

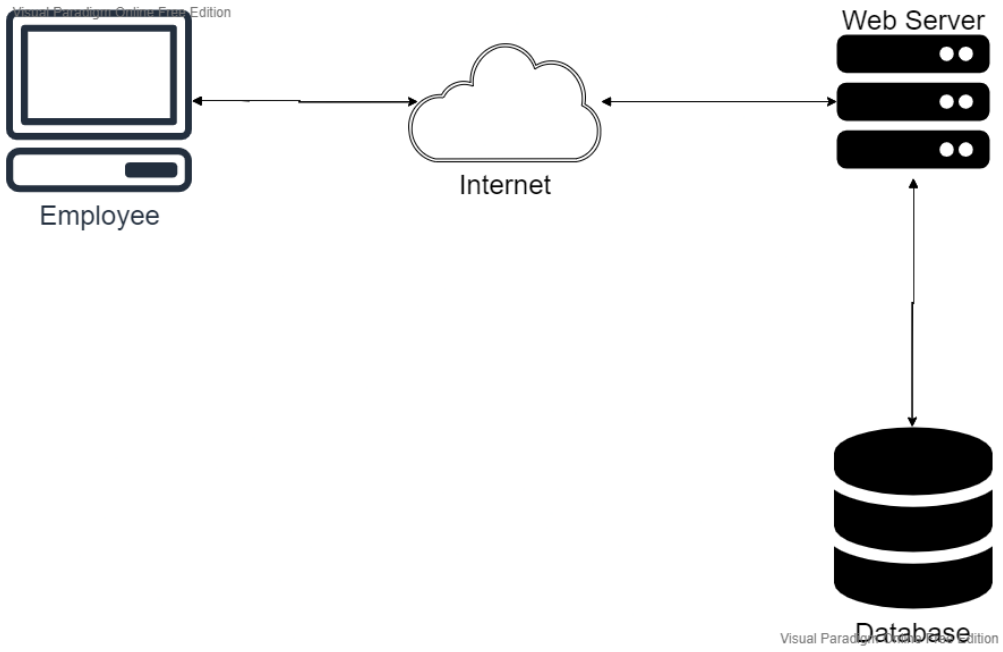


Figure 4-4 Simple architecture overview for COOP

4.6.4 Database Schema Design

Certain principles lead the way in database design. These principles include not having redundant data and ensuring all information is complete and correct. When you know that your database is properly designed, you can trust it to make strategic decisions. We have designed all the necessary databases in a way that they don't generate redundancy or deadlocks. (Noah, 2019)

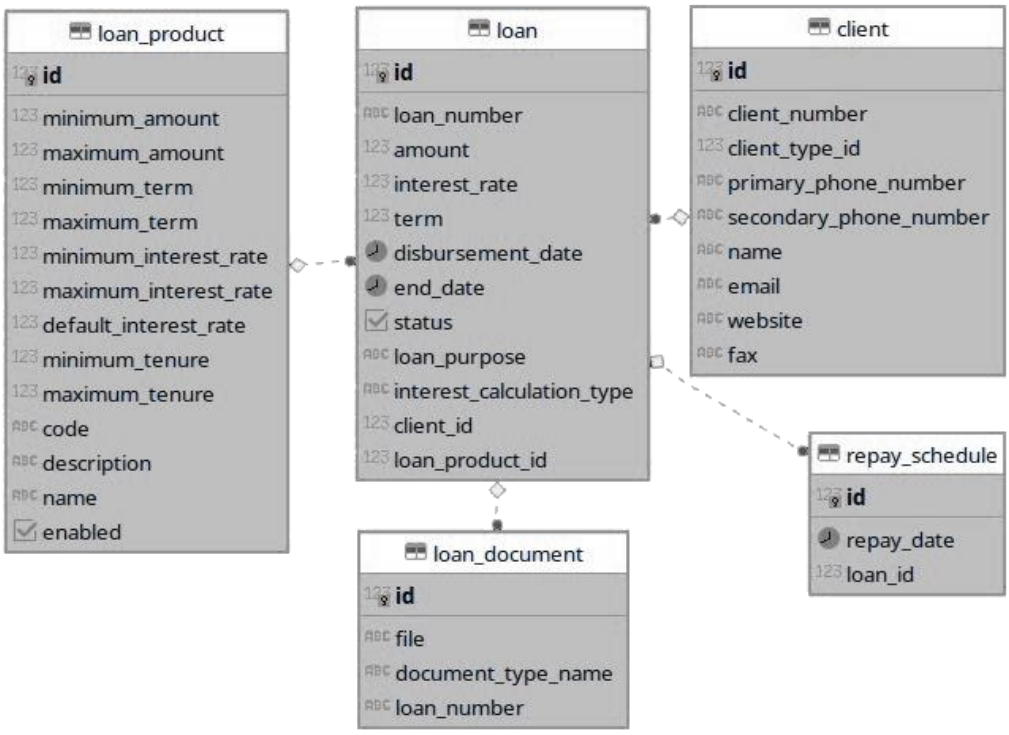


Figure 4-5 Loan database schema design

Above is the database schema for the loan entity and its related other entities. The other related entities are loan_product, loan_document, client and repay_schedule.

All of the entities have id field as their primary key. Loan Product includes different products that institute offers. Loan table holds the information about loan taken by particular client. It is linked to client and loan product table via client_id field and loan_product_id field (foreign keys) respectively.

Loan table may also associate the particular loan record with loan document and repay schedule store in their respective table. Loan document is linked to loan table through loan_number field, which is unique attribute of loan entity. And repay schedule is linked via loan_id filed i.e. primary key of loan entity.

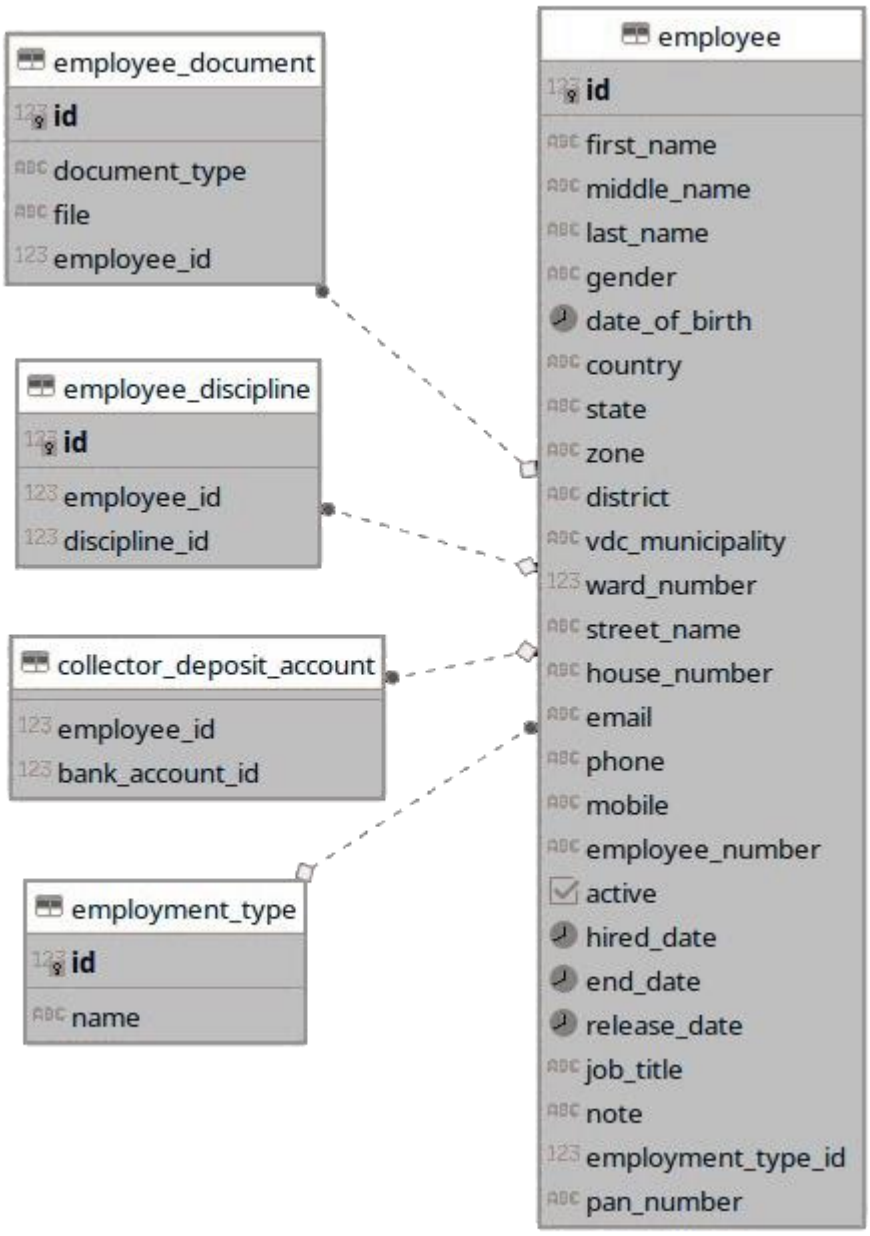


Figure 4-6 Employee database schema design

Above is the database schema of the employee entity. Its related other entities are employee_document, employee_discipline, collector_deposit_account and employment_type. Entities employee_document, employee_discipline and collector_deposit_account are linked to employee via foreign_key employee_id. And employee is linked to employment_type via employment_type_id.

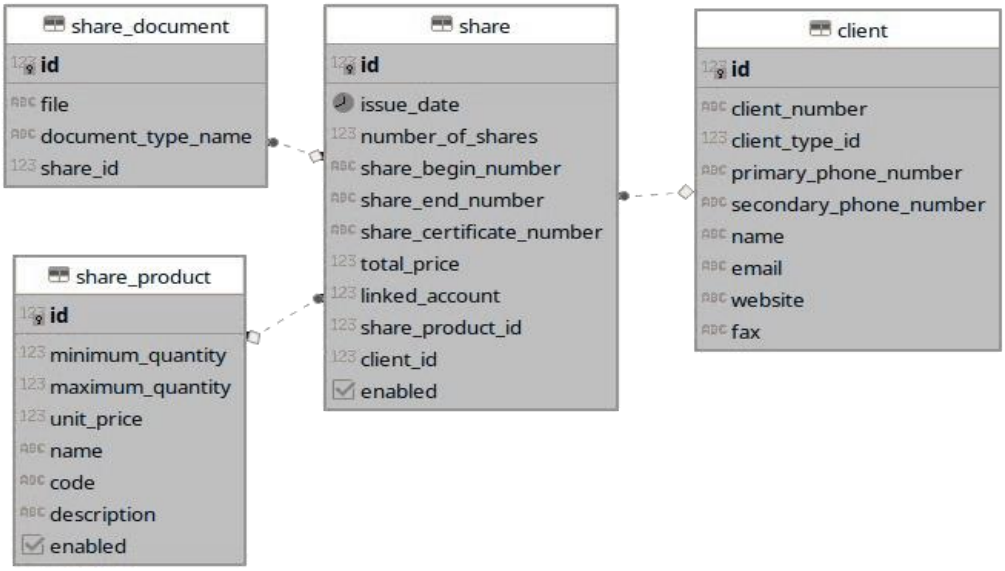


Figure 4-7 Share database schema design

Above is the database schema of the share entity. The other related entities are share_document, share_product, and client. Share holds the info about the share product owned by client. The foreign key fields share_product_id and client_id in share table links the product offered by institute with the client account. The share holded by client may also have documents which are stored in share_document table linked to share via share_id foreign key.

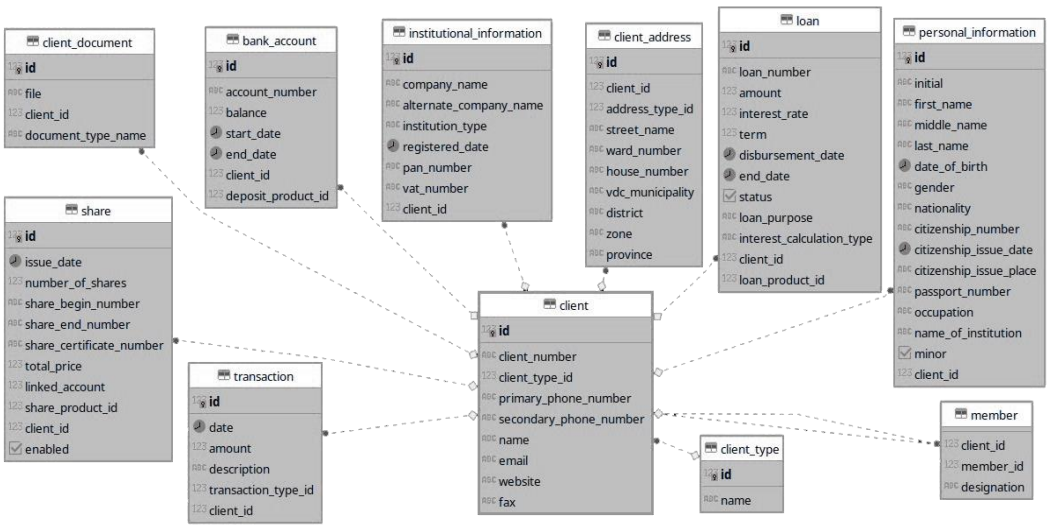


Figure 4-8 Client database schema design

Above is the database schema of the client entity. The other related entities to the client entity are; client_document, share, bank_account, transaction, institutional_information, client_address, client_type, loan, personal_information and member. Client's information is stored in client table. As in any other table, it has id field as primary key.

Client can be either Individual or Institute client which are stored in client_type table. It is linked to client via client_type_id foreign key field. Depending on type of client, their personal information or institutional information are stored in respective table linked with client_id foreign key.

Client can also have documents and address stored in respective table in which they are connected to client via client_id field as foreign key.

Furthermore, various products owned by client like loan, share and deposit(bank_account) are linked to client via client_id. And the various transaction that client performs are stored in transaction table which also contains client_id to link to client's account.

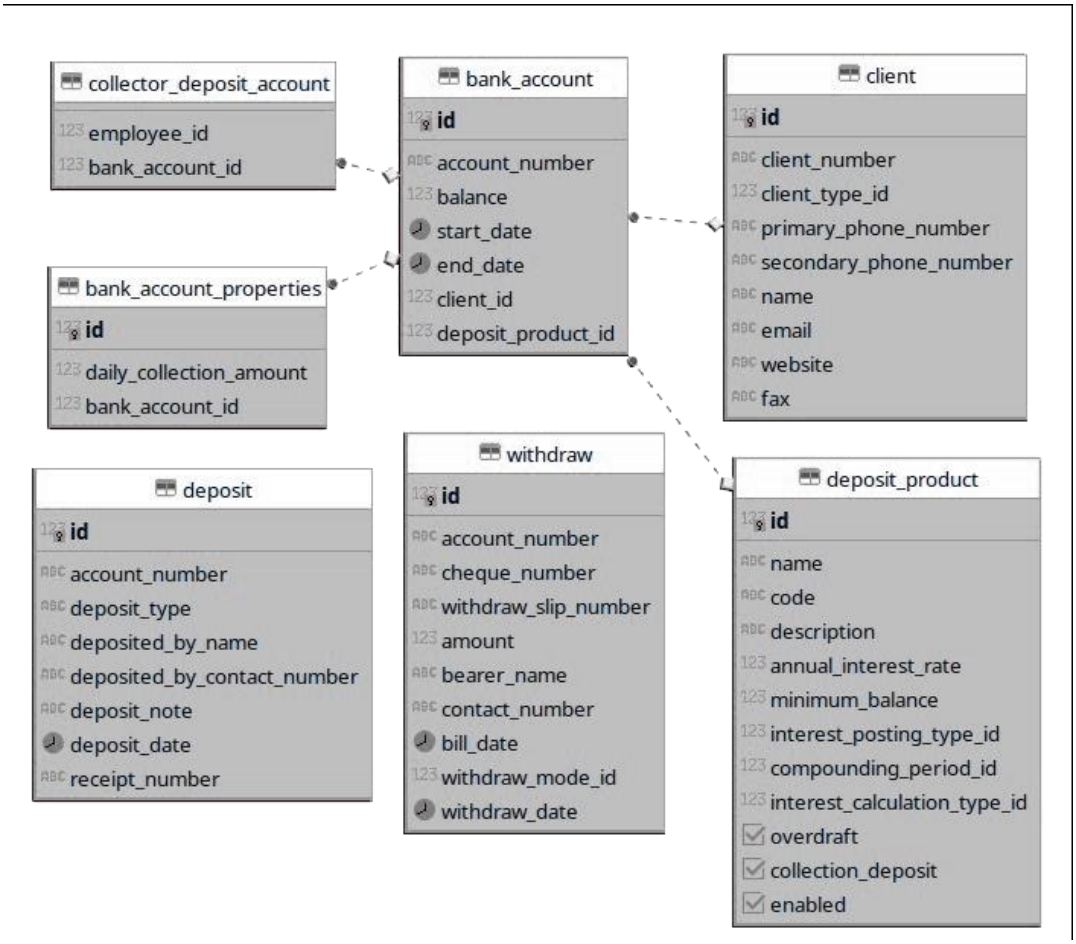


Figure 4-9 Deposit database schema design

4.6.5 User Interface Design

When designing a digital presence, whether it is a website or a mobile app, attractive graphics and engaging content are not the only things that matter. What exactly is user interface and user experience design (UI/UX Design), and why does it matter so much? Why did we look forward to have designed the system first rather than jumping into codes directly? We've covered a lot of hot topics beyond coding i.e. branding, functionality, design, integration, and usability. UX design is end-user focused from the outset that concentrates design and development effort on what users want, not on what developers think they want. As well as saving money on development costs, it stopped us from wasting internal resources. (Kambala, 2019)

It is not much long that people have started following the entire SDLC cycle or the agile process of software development sincerely. After requirement analysis and the database design, we had interface design as an integral part of our project. We used Figma software for the interface design. We often mix up UI and UX but let us tell you that it's different.

Firstly, the low-fidelity design, also called wireframes, is done. For any screen or similar screens that may come in the system, we first created basic overview of the system components layout. Wireframing helped us position the elements properly. This helped in UX, the entire customer flow and feel. Designing in such a way that users do their job in less than 3 clicks, UX part was pretty sensitive.

Figure 4-10 Low fidelity wireframe

Then comes the high-fidelity design that is the mockup or prototype. Each page is designed and connected to each other in such a way that we get to experience and see the system as a whole, as how it would look after the development is complete.

Here are two images for instance, first the low-fidelity and another high-fidelity of the login screen. The same thing has been coded later on. This is just a mockup. We may feel like this is a waste, but actually, making changes to interface design saved

us cost and time rather than making changes to codes after software is fully developed.

Figure 4-11 High fidelity prototype

Some other interface design components such as home dashboard and teller withdraw page design has been included as it becomes complicated to include the entire design process in this report.

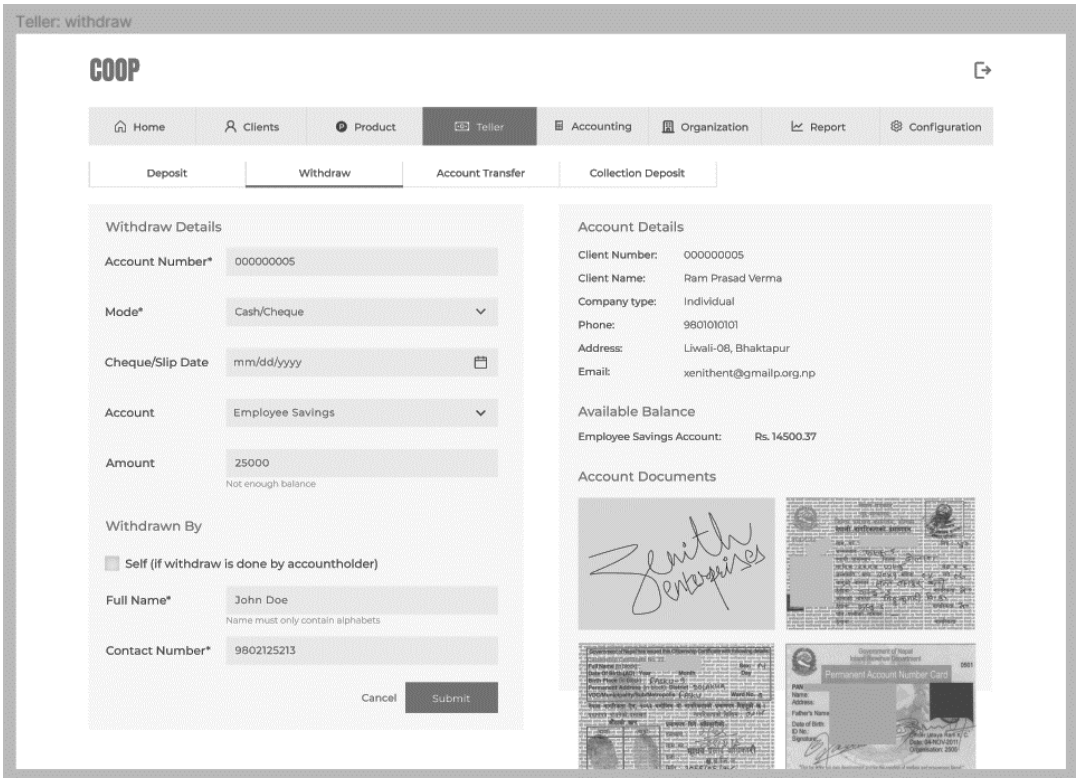


Figure 4-12 Teller withdraw page design

Just imagine a developer scratching their head to create this view without having a pre-visualized content. It becomes really hard to code from imagination on the placement of elements, how they should be treated, which elements are most important and should be shown first, etc. For example, user must be able to see total transactions made, amount of those transactions, whether it is for loan/share/deposit, etc. Outstanding loan interests, if shown, could make the user/teller be able to inform that customer quickly. Calendar is also an integral part for any financial officer. Even the charts are used, not just for filling page or making it look attractive, but so that users can take fast yet accurate decisions.

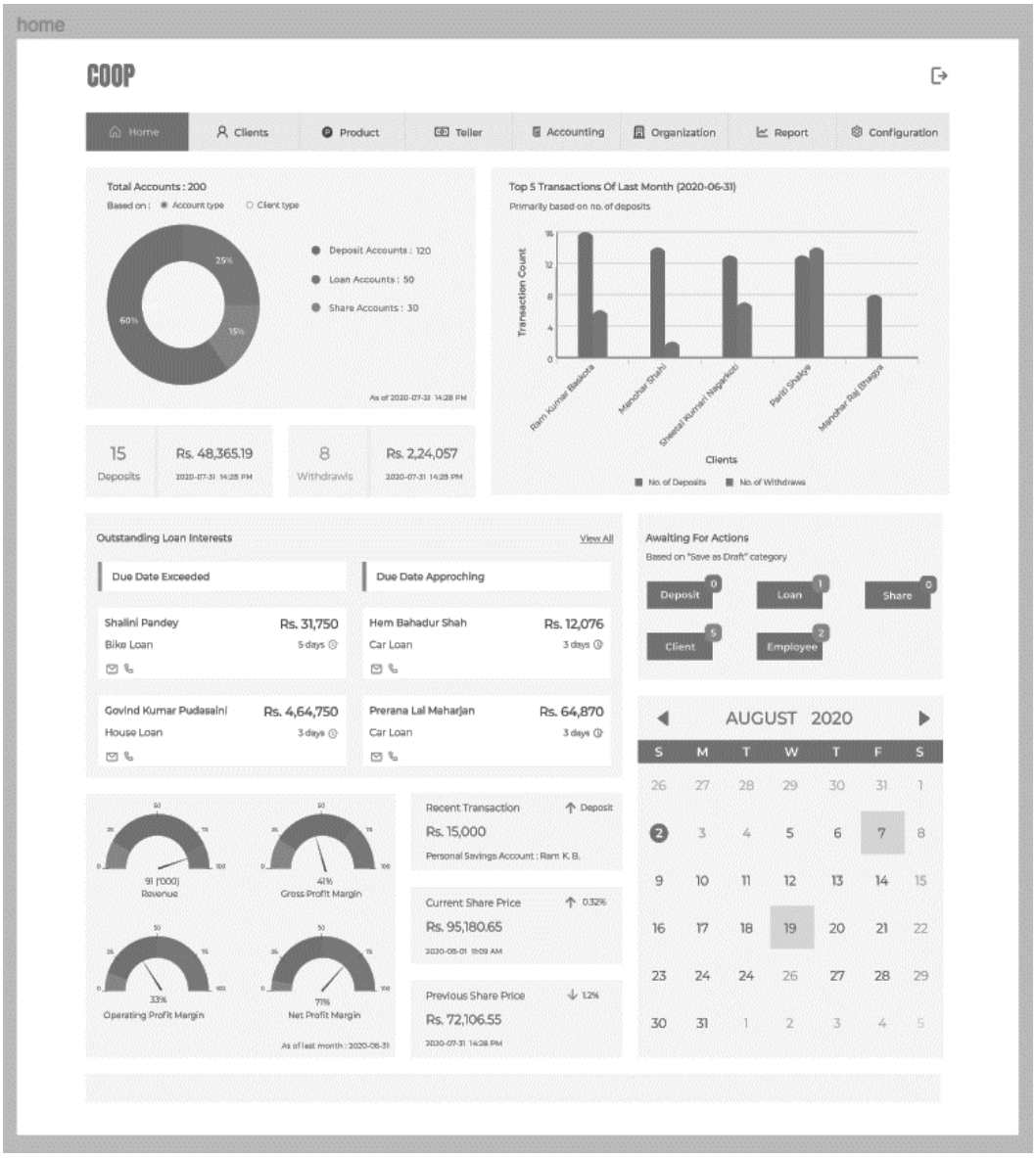


Figure 4-13 Home dashboard prototype

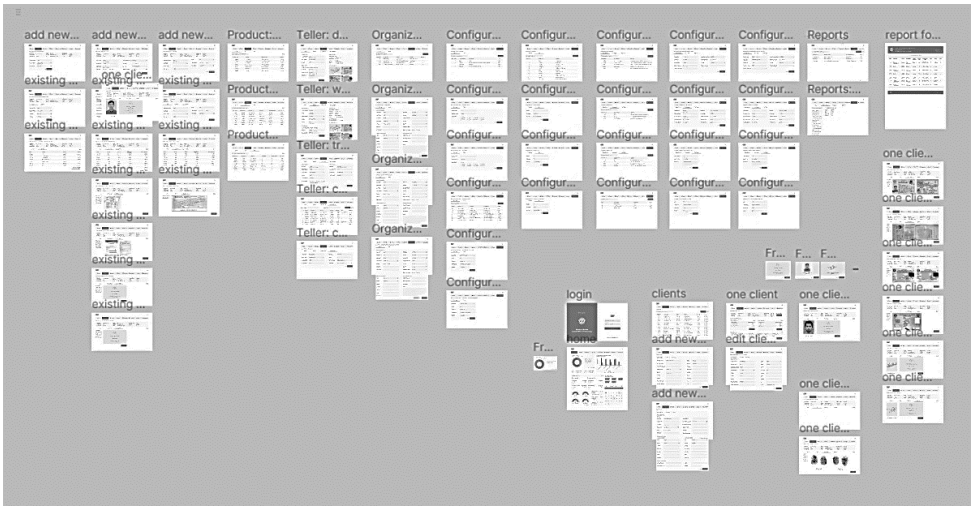


Figure 4-14 Basic prototyping overview

These images show the frames of interface design we created in Figma. All pages are designed using frames, primary color blue and minimalistic corporate look. On the other hand, there is a rough representation on how each page and related elements are linked. These prototypes may extend up to more than 100 screens but for now we only have 75+ screens designed.

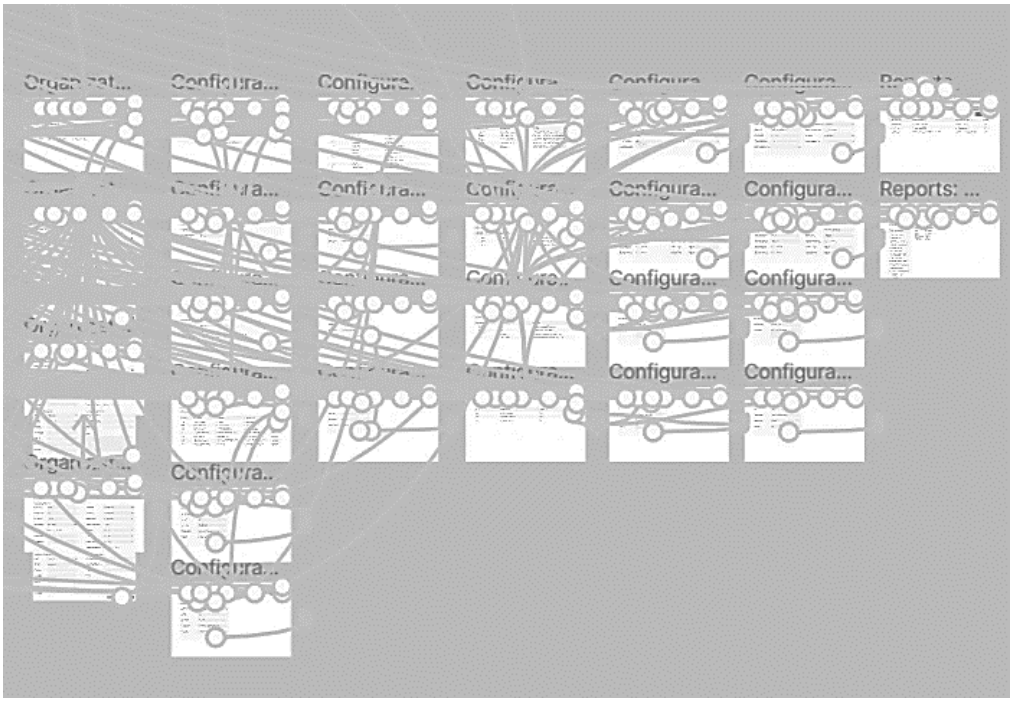


Figure 4-15 How prototype links are created

4.6.6 Sequence Diagram

The sequence diagram is used primarily to show the interactions between objects in the sequential order that those interactions occur. Not just for developers, an organization's business staff can find sequence diagrams useful to communicate how the business currently works by showing how various business objects interact, (Bell, 2004). We have included a sequence diagram to show teller operation action sequence.

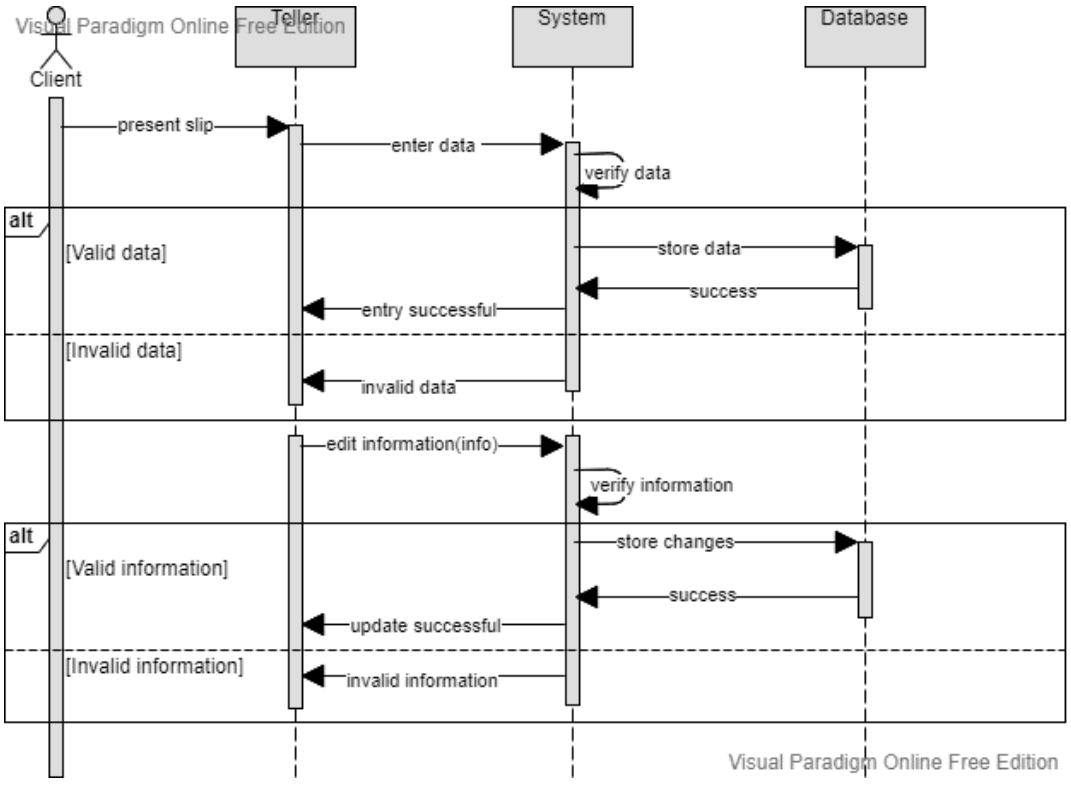


Figure 4-16 Sequence diagram for COOP teller flow

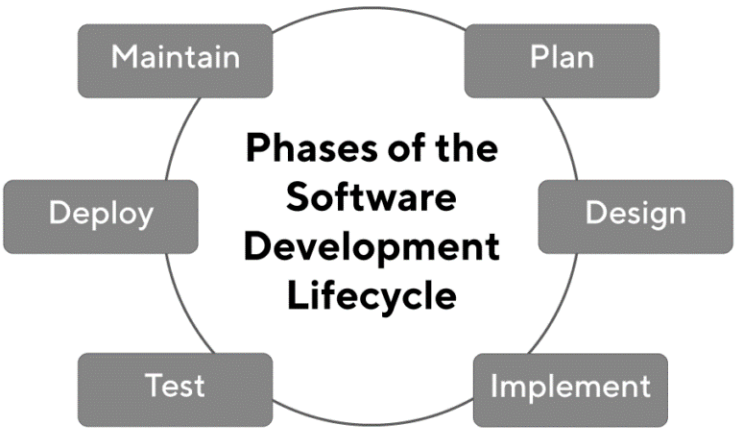
The sequence diagram represents a basic flow for teller. There are mainly four components interacting with each other- client, teller, the system and database. First, a client presents slip to the teller. The teller provides information to the system where it verifies whether the given information is valid or not. If the information is valid, then entry is successful, meaning that information is saved in the database. Teller can also update and delete the information. If the input is valid and update or delete is successful, message is prompted to the teller.

4.7 System Development Lifecycle (SDLC)

SDLC is a framework that development teams use to produce high-quality software in a systematic and cost-effective way. The SDLC methodology is used by both large and small software organizations. These teams follow development models ranging from agile to lean to waterfall and others. The software development lifecycle gives organizations a methodical, step-by-step approach to developing successful software. From gathering the initial requirements for a new product, through maintaining a mature product on the market, SDLC is important, ("SDLC Overview", n.d.).

Objectives of SDLC are as follows:

- To establish appropriate levels of management authority to provide timely direction, coordination, control, review, and approval of the system development project
- Documenting requirements and maintaining trace ability of those requirements throughout the development and implementation process
- Ensuring that projects are developed within the current and planned information technology infrastructure (Software Development Life Cycle, 2016).



(ProductPlan, n.d.)

Figure 4-17 SDLC cycle

4.8 Prototyping Model

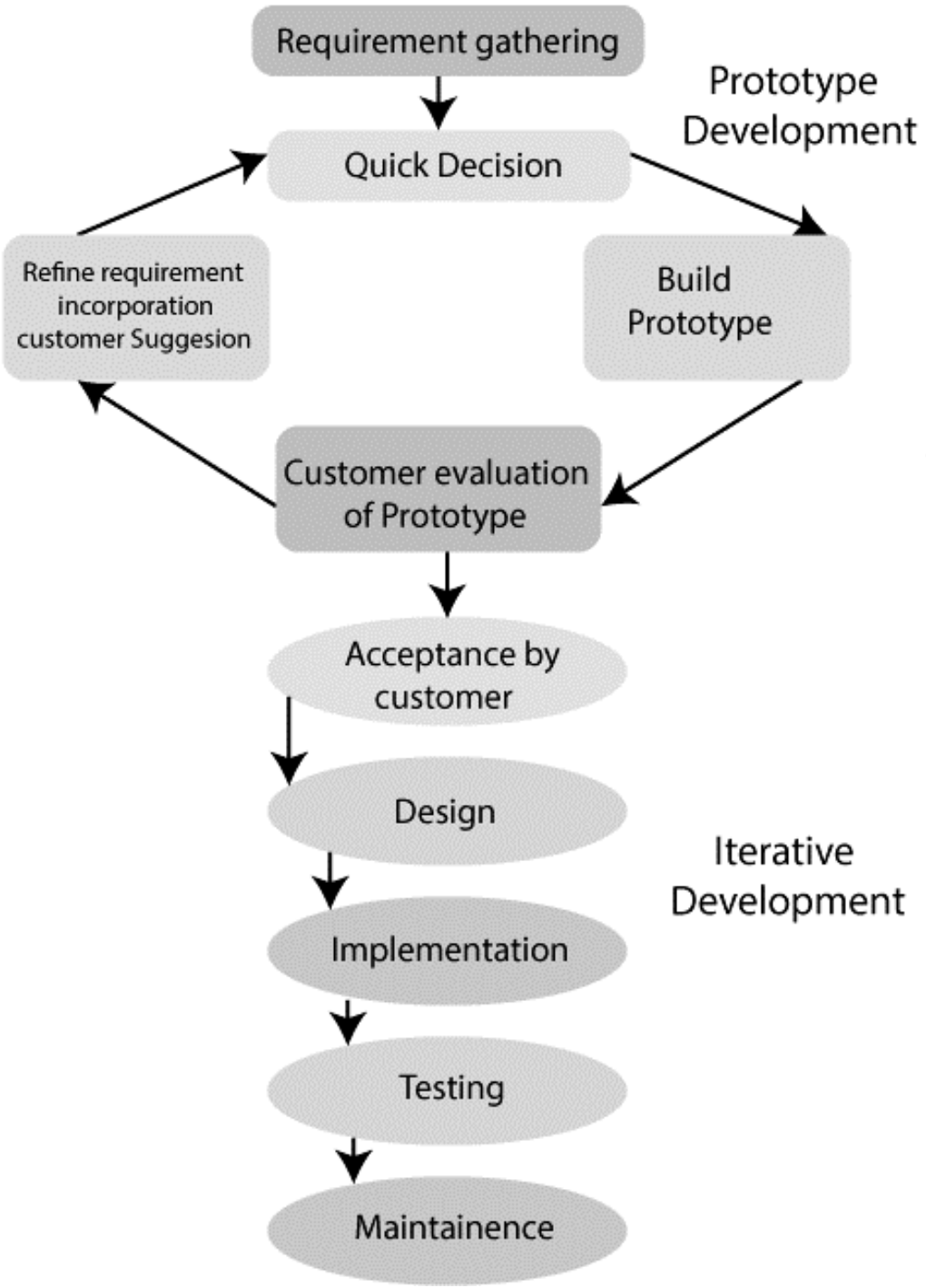


Figure 4-18 Prototyping model

Under this model, the final expected product was broken into different small pieces of prototypes and being developed individually. In the end, when all individual pieces were properly developed, then the different prototypes were collectively merged into a single final product in their predefined order. All the modules were refined carefully.

It's a very efficient approach which reduces the complexity of the development process, where the goal is divided into sub-parts and each sub-part is developed individually. The time interval between the project begin and final delivery is substantially reduced because all parts of the system are prototyped and tested simultaneously. Software Prototyping is most useful in development of systems having high level of user interactions such as online systems. Systems which need users to fill out forms or go through various screens before data is processed can use prototyping very effectively to give the exact look and feel even before the actual software is developed. Software that involves too much of data processing and most of the functionality is internal with very little user interface does not usually benefit from prototyping. Prototype development could be an extra overhead in such projects and may need lot of extra efforts.

Of course, there might be the possibility that the pieces just not fit together due to some lack ness in the development phase – this can only be fixed by careful and complete plotting of the entire system before prototyping starts.

Why did we use this model? There were few reasons of using this model under SDLC:

- If there was any missing functionality, it could be identified easily
- Requirement changes were allowed
- Due to customer approval we could find the errors at early stage
- Customer involvement was there which lead to better solutions for any confusion / complexity / difficult functions
- The developed prototype could be re-used by developer and test engineer, ("Introduction to SDLC", 2021)

The basic idea in prototype model is that instead of freezing the requirements before a design or coding can proceed, a throwaway prototype is built to understand the requirements. This prototype is developed based on the currently known requirements. Prototypes can be of multiple types as shown – low fidelity and high fidelity. Both low and high fidelity prototypes were created for this project. Prototype model is a software development model. By using this prototype, the client can get an “actual feel” of the system, since the interactions with prototype can enable the client to better understand the requirements of the desired system. Prototyping is an attractive idea for complicated and large systems for which there is no manual process or existing system to help determining the requirements, ("Prototype Model", 2019).

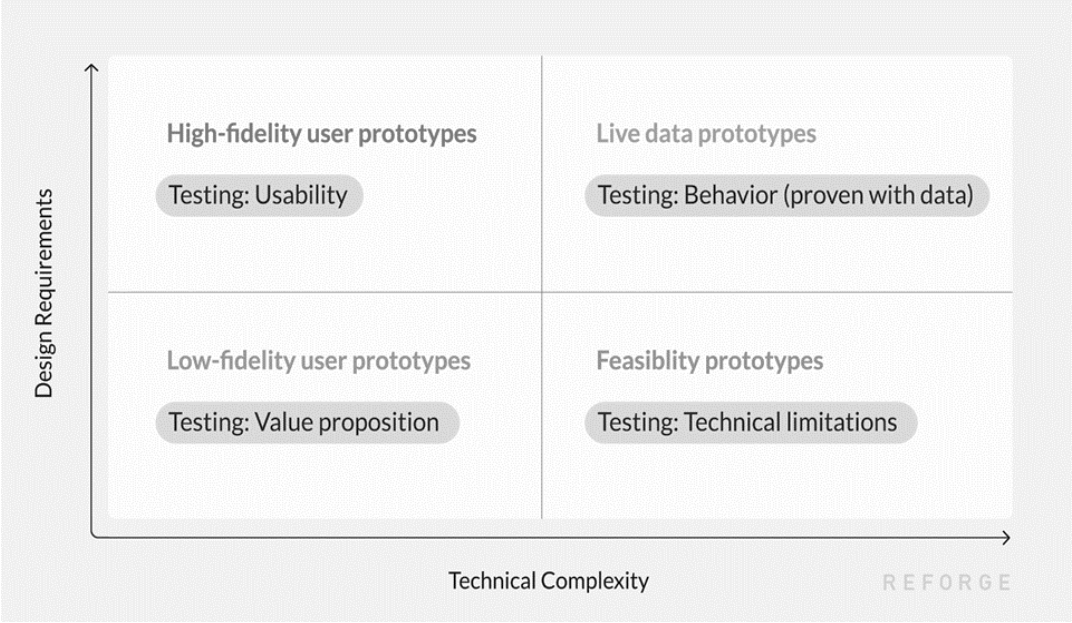


Figure 4-19 Types of prototypes

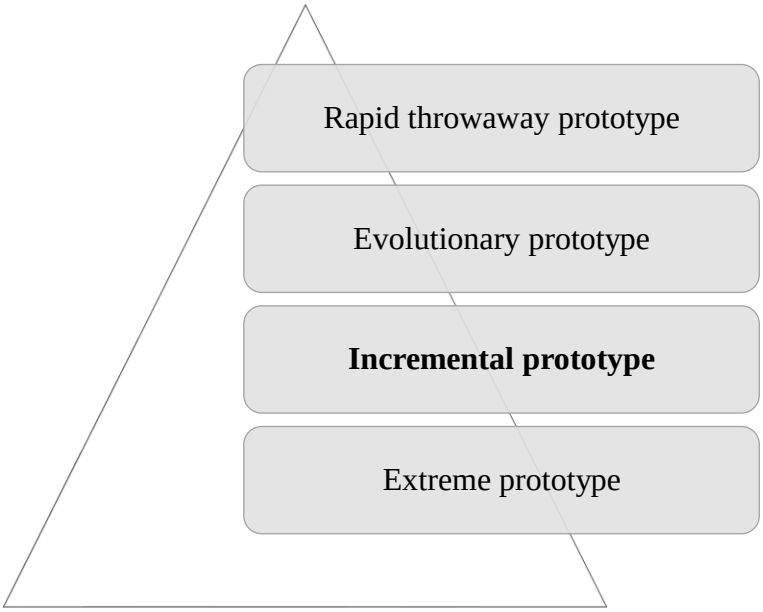
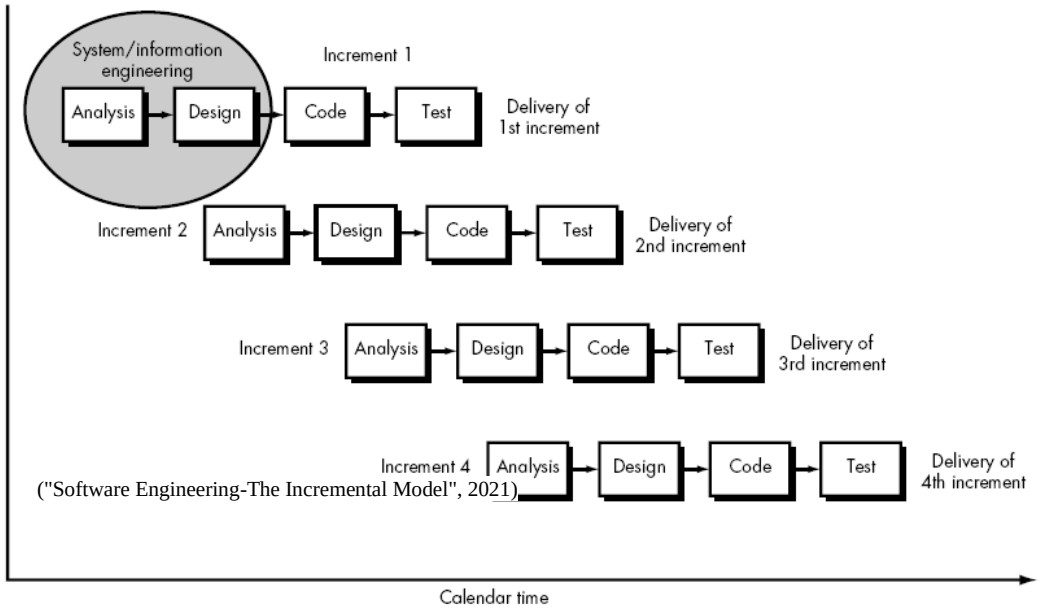


Figure 4-20 Types of prototyping



("Software Engineering-The Incremental Model", 2021)

Figure 4-21 Incremental prototyping technique

The incremental prototype technique is used for this project. In incremental Prototyping, the final product is decimated into different small prototypes and developed individually. Eventually, the different prototypes are merged into a single

product. This method is helpful to reduce the feedback time between the user and the application development team. The incremental prototyping model can be similar to the incremental model. Since it comes under SDLC, we can describe the stages in common SDLC stages and all the activities that were done in those steps are shown below.

4.8.1 Requirement Analysis

Also called planning and analysis, the first stage of new software development will be to gather all relevant information from stakeholders and analyze this information to determine what will be feasible. This includes compiling requirements, studying user personas, and agreeing on the product's purpose. During this phase, the team will also discuss the opportunities and risks of pursuing the project. Here in COOP too, the requirements were gathered, as discussed in earlier sections. They were analyzed before starting any other activities. Even requirement feasibility analysis was done and the risky parts or unachievable parts were cut off in the first stage itself.

4.8.2 Design

After confirming the requirements, we are supposed to answer "How will we build this?". Basic overview of the system is given through designs such as UML diagrams and UI/UX designing.

Initially for the designing part, Creately software is used to create a basic overview of the system. Creately supported with UML diagrams. Use-case diagrams give us a clear picture of functional requirements of the system, the actors who perform those functions and the system environment as a whole. Since the software was of trial version, a switch was made to Visual Paradigm, which did the same job. The activity diagram shows a list and flow of activities a user can perform, such as where does the user reach after logging in? User reaches the dashboard and starts data entry or access. Similarly, the schema diagram shows the backend connections of database tables and their respective attributes, ("UML Diagram Tool", 2021).

Once the system structure design was completed satisfactorily, Figma software was used to create prototypes and mockups. Starting from the possible color selection for the system, fonts appropriate for good visibility to the entire user flow mapping. Overall, the UI/UX Design is supported by Figma. This phase alone took days to complete as it is a base for the software development process to start, minimizing the risks and impacts of changes. The repayment schedule was also designed with the company logo, placement of user information, fitting in multiple columns in one page, etc. Even the prototype is tested by multiple possible users. Frequent changes were continuously made such as the position of teller withdraw section, responsiveness in all browser sizes, uploading the documents section, displaying and editing confidential documents, etc. All of these are based on the system structure design. Even though we followed agile process, the development process did not start until the prototyping was satisfactory.

4.8.3 Development

TIBCO Jaspersoft Studio is used to generate the reports according to data available such as the report for clients, employees, deposits, loan disbursement etc. The repayment schedule is a core feature of this system. It is calculated on the basis of interest rate, the principal amount, amount due and repayment time. The formulas are generated and used to calculate how much is to be paid next time. These schedules can be printed out as normal documents or saved into a csv or pdf file as required by the user. They are based on the prototypes created before. Jaspersoft provides the facility to do so, ("Reporting and embedded business intelligence software", 2021).

The main idea behind developing this project is to simplify the transaction including the attributes such as credit, debit, deposit, client, transaction, employees, interests, loan etc. making it secure and simple for users to work using IT. This organizational application not only helps to ensure the easy working of entire organizational work but also helps to ensure the secure and reliable work done. How is it actually implemented then? We planned to develop a simple usable and visible interface for users. Well, one of the main issues that we were facing is individual or bulk interest

rate change and calculation. However, we overcame it by changing them from either product section for bulk and individual profile section for individual rate change.COOP is built under a different system for frontend and backend like angular for frontend and spring boot for backend. They both are independent systems and don't know each other so API can help in establishing communication between them. So, we have created an API as part of the backend and the frontend will hit the API endpoint with different types of HTTP request methods like GET, POST, PUT, DELETE. The endpoint will return data to frontend and frontend will display the data.

MySQL is an open source relational database management system. It can store data in tables. In our system we have created many tables and have created relationships among them so that we can reduce data redundancy. Some of the names of tables are users, user_group and role. user and user_group have a many to many relationships so that one user can have many user groups and one user_group can have many users, likewise role and user_group also have a many to many relationships.

Spring boot is one of the most powerful java backend frameworks for standalone applications. It can be used to develop full backend and frontend applications but, in our application, it only serves as a backend. We have used spring boot for handling all the database operations, creating API, generating reports, validating data, generating JWT so that we can authenticate and authorize the user of the system.

Managing version of any system manually can be a headache and there is no exception for databases either so Liquibase comes in handy for maintaining the versions of the databases and helps in the migration. It can automatically generate tables from schemas, ("Liquibase Best Practices", 2021). Some data should be pre populated in the database to run the system as expected and with the help of Liquibase we can enter data to be saved in some csv files and populate the table. We have imported a third party library of maven Liquibase and we can update the database by the command "mvn liquibase:update" in the terminal. The classes in database have already been discussed in the above section. The main classes

associated with the system are Client, Product, DepositProduct, LoanProduct, Loan, Deposit, Withdraw, Employee, etc. Further, client can be of two types i.e. Institutional Client or Individual Client. The Product class defines the services that the co-operative has to offer. It could be a deposit product, loan product or share product. Likewise, disbursed loan information is defined by the Loan class.

Angular is a TypeScript based web application framework which is introduced by Google for creating a single page application, ("Angular", 2021). It supports different concepts like dependency injections, directives. We installed different third party libraries to make our app more powerful and easy to use like we have used moment for date formatting, ag-grid for tables, node-sass for SCSS and different other libraries in our application. COOP is then deployed after development to its own domain.

4.8.4 Testing

Testing is the process of checking the functionality of the software products to ensure whether they match with the expected requirements or not. Testing can identify the bugs if any, in the early stage, which can be fixed saving time and cost. So, it is a very important part of software development. Various frameworks are available for testing. TestNG is one of the testing frameworks available for writing unit and integrated test cases of java applications. It allows developers to write flexible and powerful test cases by providing functionalities like:

- Easy annotations,
- Grouping test cases
- Sequencing the execution of test cases
- Data driven testing (Parameterized)

4.8.4.1 Unit Testing

In this testing, the smallest components or the units of the software are tested. These testing are usually carried out to validate the functionality of each unit of the software. In our system, the units refer to the methods belonging to classes, (Mikhalchuk, 2020). While performing the unit testing, the method is tested

independent of the various methods and services it depends on. Hence, it allows to detect the errors and mechanism to quickly fix them. As the methods have to be tested independently, various mock/fake objects are used to assist the unit testing.

The unit test case is for validating the functionality of the controller method which is executed when firing the post API with base path + "/authenticate". As the method is depending on three other objects i.e. UserService, UserMapper and JwtTokenUtil are mocked (created fake objects) using Mockito Library. We don't actually execute these dependencies, rather mock the functionality of those services. It helps us to focus on testing a single method. Then the actual result is stored by executing the method and is checked for the existence of user and token which is at last asserted with expected existence of user and token. The data passed to the API call as well as the expected result are fetched from the parameter and are passed to the test function from the data provider method defined in the annotation.

In the data provider method, data can be returned as the array of objects. Each object can represent an individual test dataset. It can be explained as:

- In first dataset, body content exists with valid user, so we expect 200 status and response with user and token
- In second, user is invalid hence we expect 403 status without user and token
- Lastly, with empty body, we don't expect user or token in response

4.8.4.2 Integration Testing

After testing the individual units, the functionality of these units are to be tested as a single group. It is done to validate the performance of a software system as a whole. Fully functioning units could malfunction when used together. Hence, it verifies the correctness of the data transferred from one unit to another.

The test cases are for getting the employee data and adding new employee data. The second method depends on the testGetEmployees method. It means that the 2nd test case will be executed after the 1st test case.

In the testPostEmployee test method, we are creating the mock post request to employeeUrl. jsonBody contains the employee info passed from dataProvider

below. When the API is fired with given data, we are expecting the status to be Created i.e. 200 and asserting the firstName of the employee returned to be “Lucifer”, which is the name that we sent in request.Unlike the unit testing, the integrated testing will not mock the services the particular unit depends on, but will test the correctness of the data being passed between those units. Hence, once the API is fired with appropriate data, it will create the new employee and return the new employee.

4.8.4.3 System Testing

Once unit testing of every component is performed and the code components are also combined and tested together during integration testing, the system as a whole is tested for completeness with respect to requirements. Manual testing has been carried out by trying out the system functionality. Two pass and fail test cases for system testing is shown below:

Table 4-1 System testing - test case 1

Test case ID	Test case description	Pre-conditions	Test steps	Test Data
TC_01	Check response when valid data are entered.		1. Navigate to withdraw page.	
		Valid account number	2. Enter Account number.	Account Number: 100200100
			3. Select the mode of withdrawal.	Mode: Cheque
			4. Enter the cheque/slip number.	Check number:1234
			5. Enter cheque/slip date.	Cheque/Slip date: 08/31/2020
		Valid amount	6. Enter Amount.	Amount: 50000
			7. Check the box if the withdrawal is done by the account holder.	Self: Checked
			8. Enter the name.	Name: Nischal Kumar Shakya
			9. Enter contact number.	Contact: 9841234563
			10. Click on withdraw button.	

Expected Result	**Actual Result**	**Status**
Able to see the withdraw page.	As expected	Pass
Credential can be entered and account holder information is displayed.	As expected	Pass
Mode can be selected.	As expected	Pass
Credential can be entered.	As expected	Pass
Credential can be entered.	As expected	Pass
Credential can be entered.	As expected	Pass
Box should be checked.	As expected	Pass
Credential can be entered.	As expected	Pass
Credential can be entered.	As expected	Pass
A Dialog box should pop-up with a message: Withdraw detail saved successfully.	As expected	Pass

Table 4-2 System testing - test case 2

Test case ID	Test case description	Pre-conditions	Test steps	Test Data
TC_02	Check response when invalid account number is entered.		1. Navigate to withdraw page.	
		Invalid account number	2. Enter Account number.	Account Number: 100200900
			3. Select the mode of withdrawal.	Mode: Cheque
			4. Enter the cheque/slip number.	Check number:2344
			5. Enter cheque/slip date.	Cheque/Slip date: 08/31/2020
		Valid amount	6. Enter Amount	Amount:40000
			7. Check the box if the withdrawal is done by the account holder.	Self: Checked
			8. Enter the name	Name: Nischal Kumar Shakya
			9. Enter contact number	Contact: 9841234563
			10. Click on withdraw button	
Expected Results			**Actual result**	**Status**
Able to see the withdraw page.			As expected	Pass
Credential can be entered but account holder information is not displayed.			As expected	Pass
Mode can be selected			As expected	Pass
Credential can be entered.			As expected	Pass
Credential can be entered.			As expected	Pass
Credential can be entered.			As expected	Pass
Box should be checked.			As expected	Pass
Credential can be entered.			As expected	Pass
Credential can be entered.			As expected	Pass
Failed message should be displayed.			A Dialog box popped-up with a message: Withdraw detail saved successfully	Fail

4.8.5 Maintenance

Any software developed using agile methodology continuously adapts changes. Be it bug fixing or adding new features, removing outdated information, dealing with user roles, etc. they must efficiently be supported by the system. COOP, too, needs many changes, such as adding a new feature of generating tally balance sheets, etc. But before that, it must be maintained timely. Frequent database versioning may be required. The user roles and assigning responsibility to each role must be maintained. data redundancy should be inspected. Bugs reported by users must be solved immediately. So, the team must always be available to maintain the system. Today this system may boom but tomorrow when a completely new technology is grabbing the market, we should be able to change according to that platform, may be a new framework, new language, automation, new way of data handling, etc. (White, 2006). This also counts as future maintenance, which we can successfully attain, if required.

On the other hand, the users of this system, cooperative organizations, must be provided with some training before they are handed over with the system completely. Overall working flow of the system is to be explained to the users such as teller, loan manager, and everyone associated with the system. Which section is authorized to which user role, who can access the documents, how to generate reports, how to view the dashboard KPIs and other reports, etc. must be documented and explained like a user manual? It may feel like a hassle for them to dig into so many features we have provided. But because of the simple UI, they can adapt in no time.

4.9 Activities Done

COOP is a web application which will be used by cooperatives. It eases the process of transactions of cooperative. This application consists of many modules like loan, share, deposit, withdraw, clients, employee, etc. The system can be used to handle transactions like loan withdraw, deposit collection, request handling, employee management.

As a UI/UX designer intern, the main task given to me was, obviously, to handle both UI and UX. No system will be successful unless it satisfies its users. We has a team of SDEs, PM and QA engineer. Each people had the tasks divided into some time. We initially used MS Teams to communicate and used JIRA as project management tool. I was supposed to frequently have meetings with the PM and update JIRA as well as daily intern worklog. The tasks that I was assigned with were:

- Study of existing systems in the market such as Musoni
- Study of current workflow in the cooperatives
- Research on user friendly interfaces and design practices
- Study the working of corporate web apps
- Study client and user behaviors in simulated cooperative environment
- Requirement gathering and discussions for the system
- Frequently report to the mentor
- Update the daily worklog xlsx
- Create low fidelity designs (wireframes) for all modules
- Create high fidelity designs (mockups) according to wireframes
- Work with breadcrumbs and have idea of routing
- Interact with PM and SDEs regarding what design changes are need to be done
- Design for several reports such as loan repayment schedule, dashboard reporting, error reports
- Deliver a full-fledged working prototype
- Test the flow, handle overall UX part
- Present the designs to all team members

For the UI designs, I had to consult with my mentors about current corporate design trends in the market. Along with the color combinations, icon selections and font for the system, I was responsible for creating minimalistic yet achievable and usable screens. There are more than 80 screens designed for the system. Since some design components seemed heavy and some unachievable with code, I had to frequently make changes. The most interesting part was dashboard KPI. I didn't have to handle

the slicing pat. However, UX was challenging. There are so many elements in the dashboard, that it was challenging to place those elements in correct order or even design well for visibility. Whether to place calendar on top right-hand side with sticky position or not? Is it really that important? Do users need to check calendar all the time? If not, which components will the users frequently use? Some similar questions I had to deal with were:

- What kinds of charts would the users need to make decision?
- How many clicks will a user require to achieve their purpose of logging in to the system?
- Do the report format match corporate printed reports?
- Are the colors suitable for daily use?
- Is the app becoming too fancy with all the document upload sections and is the user role management section divided properly?
- Do we really need that dialogue box or could we use a new page for filling forms?
- What really creates a USP to our system while there are other existing systems in the market?
- Is the flow correctly designed? What if a user can not find the edit profile section at all?
- Are breadcrumbs necessary for the web app?
- Why are we not designing for mobile app or even web responsiveness? Is web app only the solution?

Since we were working from home because of the pandemic, we were obliged to have daily stand-ups twice. There we discuss tasks for the day and at the end of the day, we all discussed about the day's works, issues, milestones achieved, etc. Not just a role of designer, I was also assigned to conduct meetings, communicate everything with team members, be sure about the progress and providing green light for starting the development process. Besides that, I also had the responsibility to do different design testing such as:

- Usability testing
- Visibility testing
- User interaction testing

The internship not only taught technical skills but managerial and human skills too. Having empathy for users, I had to test the system design from their point of view. So, one of the main responsibilities as a UI/UX designer was to be sure that most risky changes occur in the design phase itself. And the purpose of designing phase in SDLC is the same. Developers need not scratch their head to fix bugs or completely transform a module into something else. I was also assigned to research and create mockups for promotional website for COOP.

I also handled the documentation part for requirements analysis, development sprints for the system, even feasibility so as the developers would find it easy, and later on, be a user to the system itself so as to test the overall working of the system. Some of these are described below:

- Daily Meeting

  It is one of the core components of agile methodology. Our team attended the daily standup meeting twice a day explaining what to do and what have been completed in the day limit. In daily meetings, we basically give short update about the task that we completed, the task that you will work on in following hour, and if there is any problem that you have been stuck on.

- Sprint Turnover

  In the agile, design and development is divided into number of sprints which are usually about 2 weeks long. After each two weeks, the last spring comes to a close, and the new sprint is planned. If any tasks from previous sprint is leftover then they are carried over to the next sprint.

- Retrospective

  Retrospective is the way of improving the development process. It is one of the vital components of agile methodology. In this, the team talk about three basic

points: what went good in this sprint, what went bad in this sprint, any possible improvements?

- Live Demonstration

The demonstration of each iteration's final production is done to show the progress of the system. This allows the team and the other concerned stakeholders to be updated about the product and its progress after each iteration.

Apart from these, I was assigned to keep track of the progress I have been making to the system. Even if it was an agile development process, I had to make sure that developers could confidently code the system.

Chapter 5

SUMMARY AND CONCLUSION

5.1 Findings

After 3 months of internship at ITGlance, I am aware of so many unlearnt topics. From the basics and principles of designing websites and apps to applying those in professional corporate projects, I learned to value time, research and opportunities. The internship program not only introduced me to the real working environment but it also provided me a platform to grow professionally right at the end of undergraduate program.

- Differences in the theoretical knowledge in college and practical working
- Coping up within real working environment
- Working procedure in IT Company and procedures of getting projects from the clients
- Understood importance of coordination and cooperation in working environment
- Knowing the development procedure of effective and efficient system for any organization
- Skill to communicate and report to those parties in time regarding any kind of issues that are dealt in the organization
- Brief insight of career path

5.2 Conclusion

Internship provided a wonderful pool of opportunities for us to learn. It gave us a platform to judge our skills and our capabilities to apply the learnt knowledge into actual practice. As a part of partial fulfillment of requirement of the degree of BIM, I had to be involved in ITGlance to gain the practical knowledge and experience through the internship program. Being a part of the company as an intern has truly been a learning experience to remember. Staying in schedule, maintaining social

relationships with other employees etc. can be said the major learning from the valuable two months period of internship period. With this industrial attachment assignment, we are going through the development phase from student to amateur and in the direction of professionals.

5.3 Recommendation

Currently there is limited KPI to support in complex accounting and financial operations and decisions. This could be considered as a future work. For proper promotion and to let users know about the existence of this system, we may create a promotional marketing based informative website which will persuade users to subscribe to this system plans. The system is developed under an OOM where a large program is divided into smaller modules, (Guimarães, 1995). Hence, adding and changing the existing modules of the system would not be a difficult task. The system can be modified as per the requirement. So, any new requirement can be easily implemented in this system. The system can track the daily transactions but the accounting of financial transactions is still to be done. Another feature that can be added is sending notification to the clients about their due dates or some other events through the application. The system could work equally well in mobile app or tablet versions, rather than being limited to web app. Another feature that can be added is sending notification to the clients about their due dates or some other events through the application.

As for the internship experience, everything was almost good. However, mentors had their own jobs to focus on, which made interns feel lost at work. Also because of the pandemic, we were forced to be at home, which did not prevent us from virtual communication but lost a chance to socialize well. Mentors could have set some meet programs with all the team members before starting the work. And due to the limited time constraints, some of the features are left to be added for the efficiency of the system that was just discussed earlier.

BIBLIOGRAPHY

- Angular. (2021). Retrieved February 2021, from https://angular.io/
- Bell, D. (2004). The sequence diagram. Retrieved March 2021, from https://developer.ibm.com/technologies/web-development/articles/the-sequence-diagram/
- Bhurtyal, B. (2010). Nepal ICT Policy Challenges and Opportunity. Presentation.
- Co-op - Definition of a Cooperative Business. (2021). Retrieved March 2021, from https://ncbaclusa.coop/resources/what-is-a-co-op/
- Department IT Strategic Goals. (2010). Retrieved March 2021, from https://2009-2017.state.gov/m/irm/itplan/264054.htm
- Dhungel, R. (2019). IT History and Development in Nepal. Presentation.
- Guimarães, J. (1995). The object oriented model and its advantages [Ebook]. Retrieved from https://www.researchgate.net/publication/220391668_The_object_oriented_model_and_its_advantages
- Introduction to SDLC. (2021). Retrieved April 2021, from https://www.subjectcoach.com/tutorials/detail/contents/introduction-to-software-development-life-cycle-sdlc/chapter/prototype-model-of-sdlc
- ProductPlan. What is SDLC? [Image]. Retrieved from https://www.productplan.com/learn/software-development-lifecycle/
- Reporting and embedded business intelligence software. (2021). Retrieved February 2021, from https://www.jaspersoft.com/de
- Liquibase Best Practices. (2021). Retrieved 2021, from https://www.liquibase.org/get-started/best-practices
- Noah. (2019). Database Design [Blog]. Retrieved from https://www.arkware.com/2019/09/12/importance-of-good-database-design-and-7-steps-to-achieve-it/
- Non-functional Requirements. (2021). Retrieved 2021, from https://www.altexsoft.com/blog/non-functional-requirements/

- Prototype Model. (2019). Retrieved April 2021, from http://tryqa.com/what-is-prototype-model-advantages-disadvantages-and-when-to-use-it/
- UML Diagram Tool. (2021). Retrieved March 2021, from https://creately.com/lp/uml-diagram-tool/
- SDLC Overview. Retrieved March 2021, from https://www.productplan.com/learn/software-development-lifecycle/
- Soft, Y. (2019). History of IT in Nepal. Retrieved March 2021, from https://medium.com/@ityashri/history-of-it-in-nepal-183591c5d681
- Software Architecture. (2021). Retrieved March 2021, from https://www.synopsys.com/glossary/what-is-software-architecture.html
- Michigan Tech Information Technology. (2016). Software Development Life Cycle [Ebook] (2nd ed.). Michigan. Retrieved from https://www.mtu.edu/it/security/policies-procedures-guidelines/system-development-lifecycle.pdf
- Software Engineering-The Incremental Model. (2021). Retrieved April 2021, from https://www.1000sourcecodes.com/2012/05/software-engineering-incremental-model.html
- Technical Writing. (2021). Retrieved March 2021, from https://courses.lumenlearning.com/alamo-technicalandbusinesswriting/chapter /unit-4-b_feasibility-report_readings-2/
- White, W. (2006). Procyclicality in the financial system: do we need a new macrofinancial stabilisation framework? [Ebook] (pp. 5-6). Basel, Switzerland: Bank for International Settlements. Retrieved from https://www.bis.org/publ/work193.pdf

APPENDIX

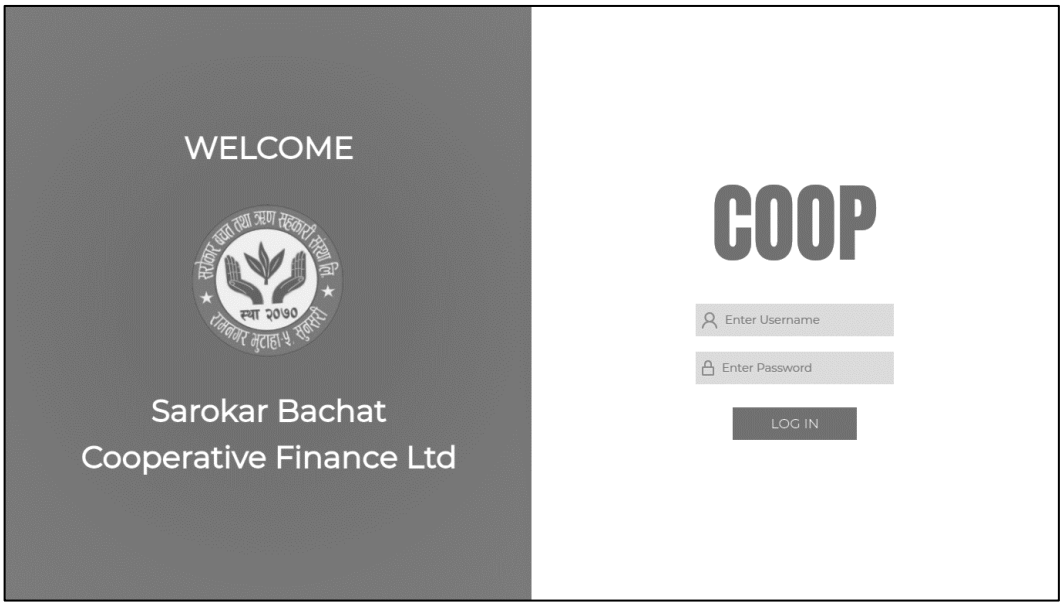


Fig: Login page for COOP system

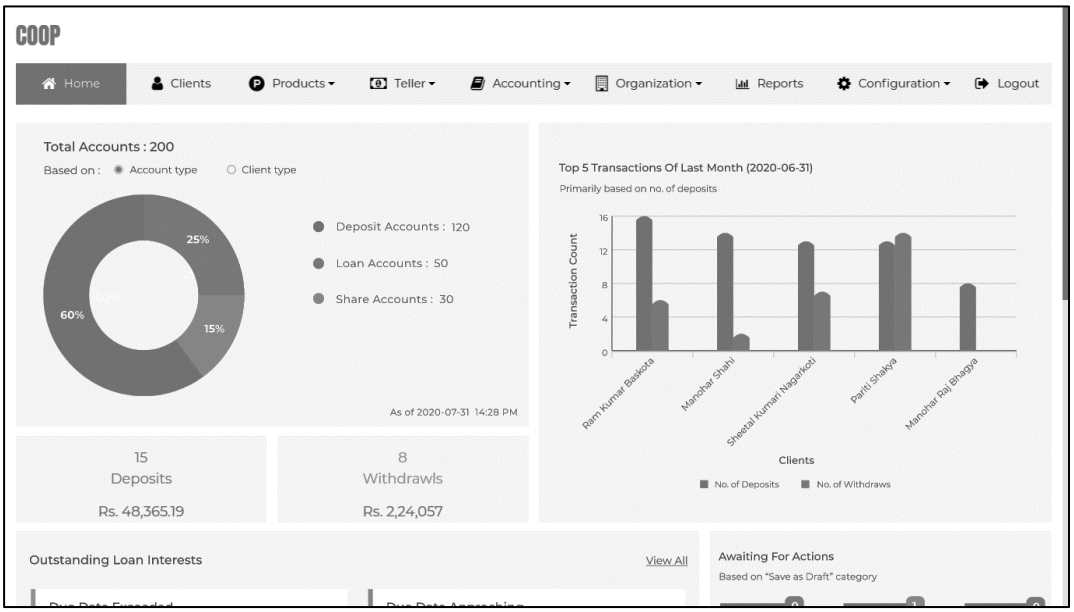


Fig: Home (dashboard) after logging in

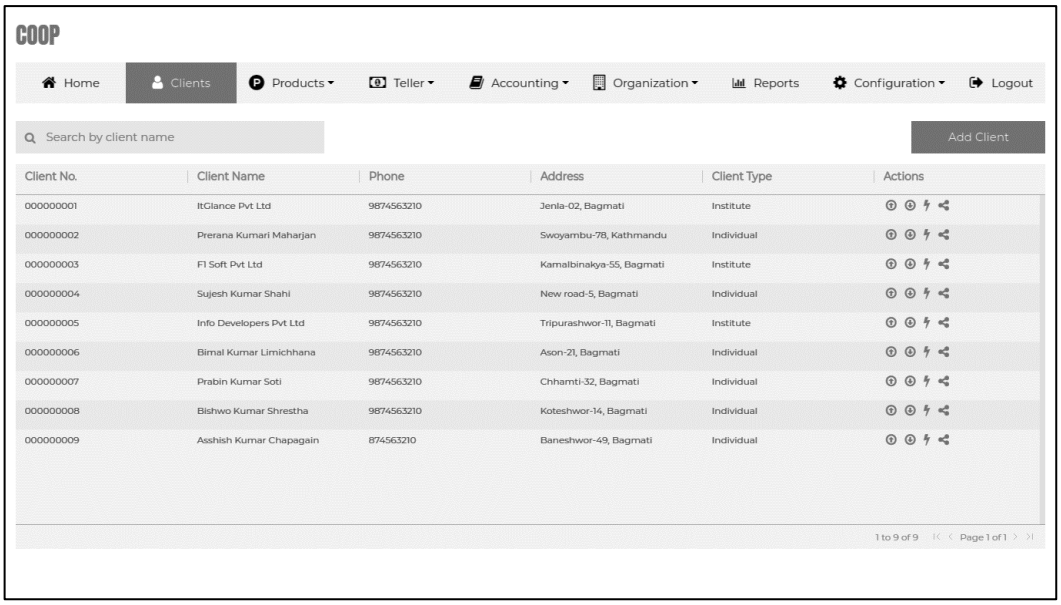


Fig: Clients page

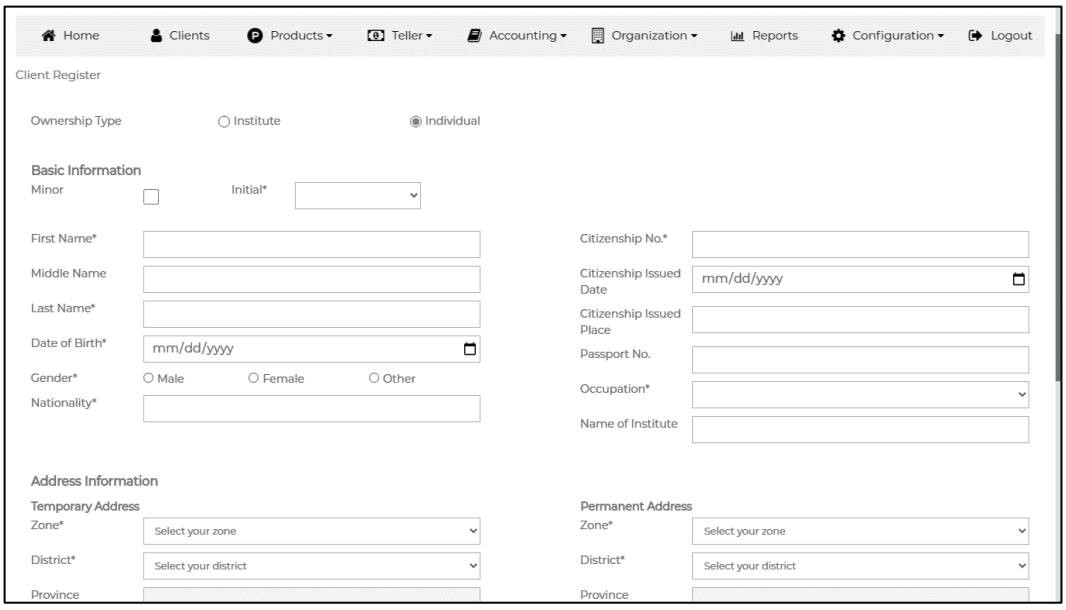


Fig: Add individual client page

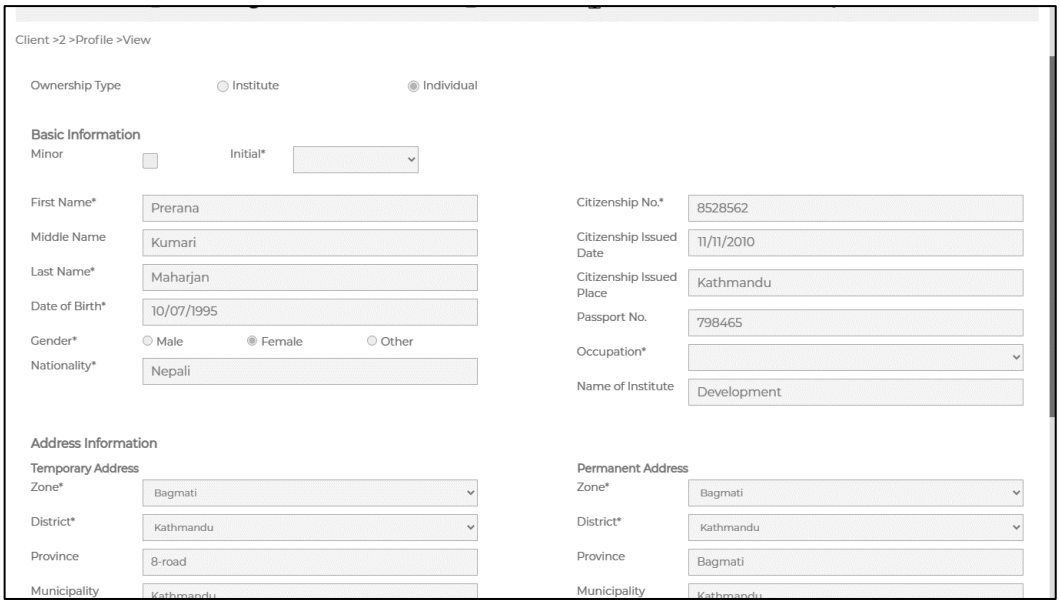


Fig: Add institutional client page

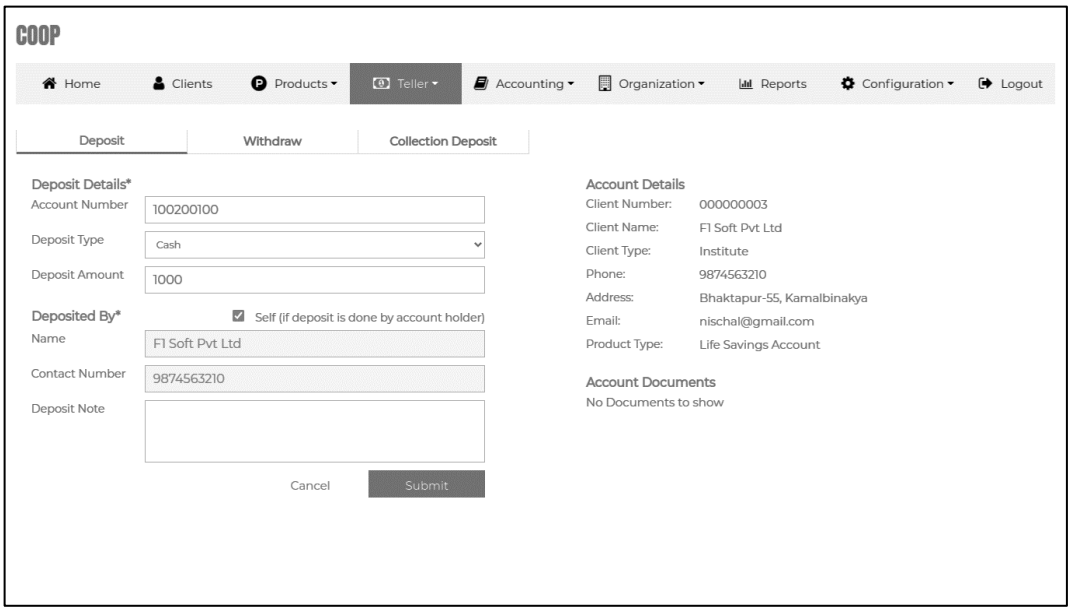


Fig: Teller (deposit) page

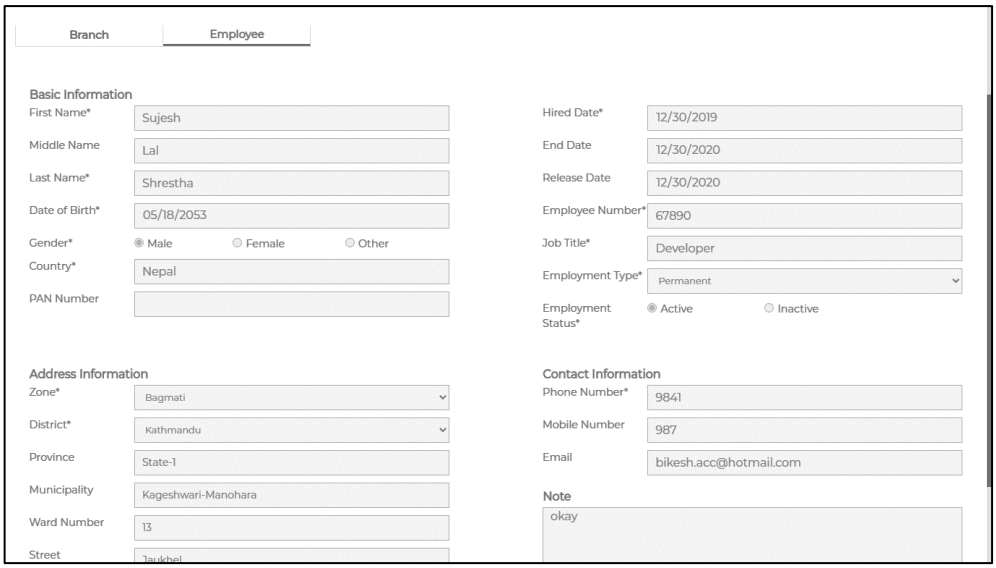


Fig: Edit employee details page

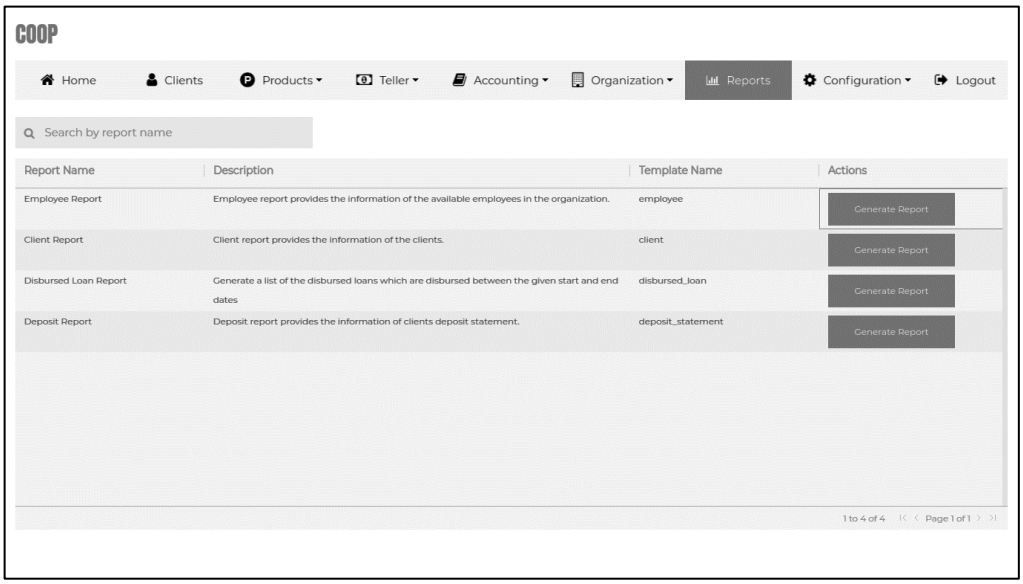


Fig: Reports page

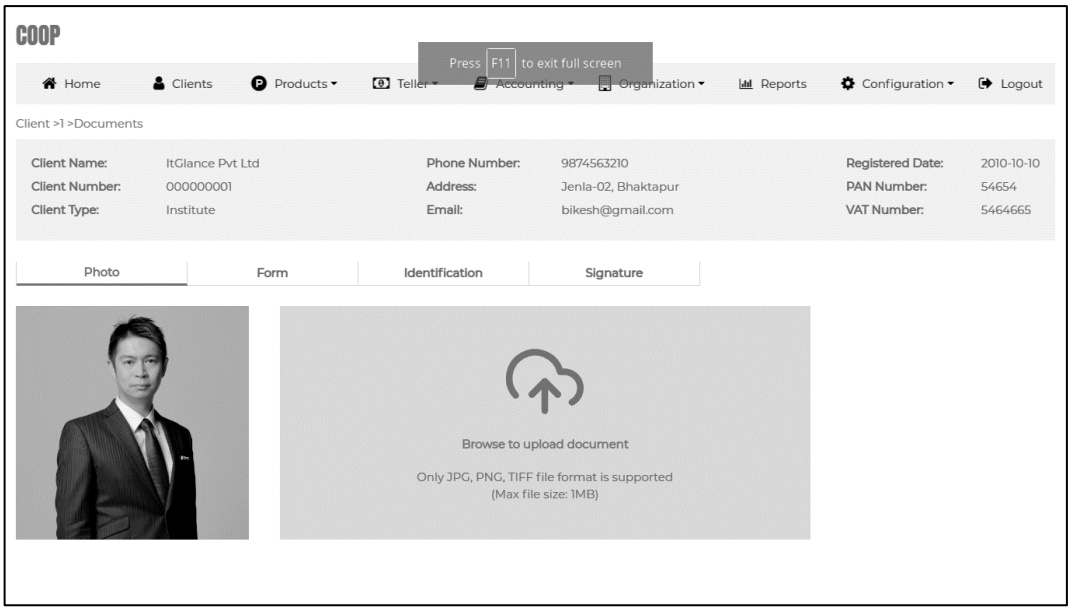


Fig: View one client page

Fig: Generated report format

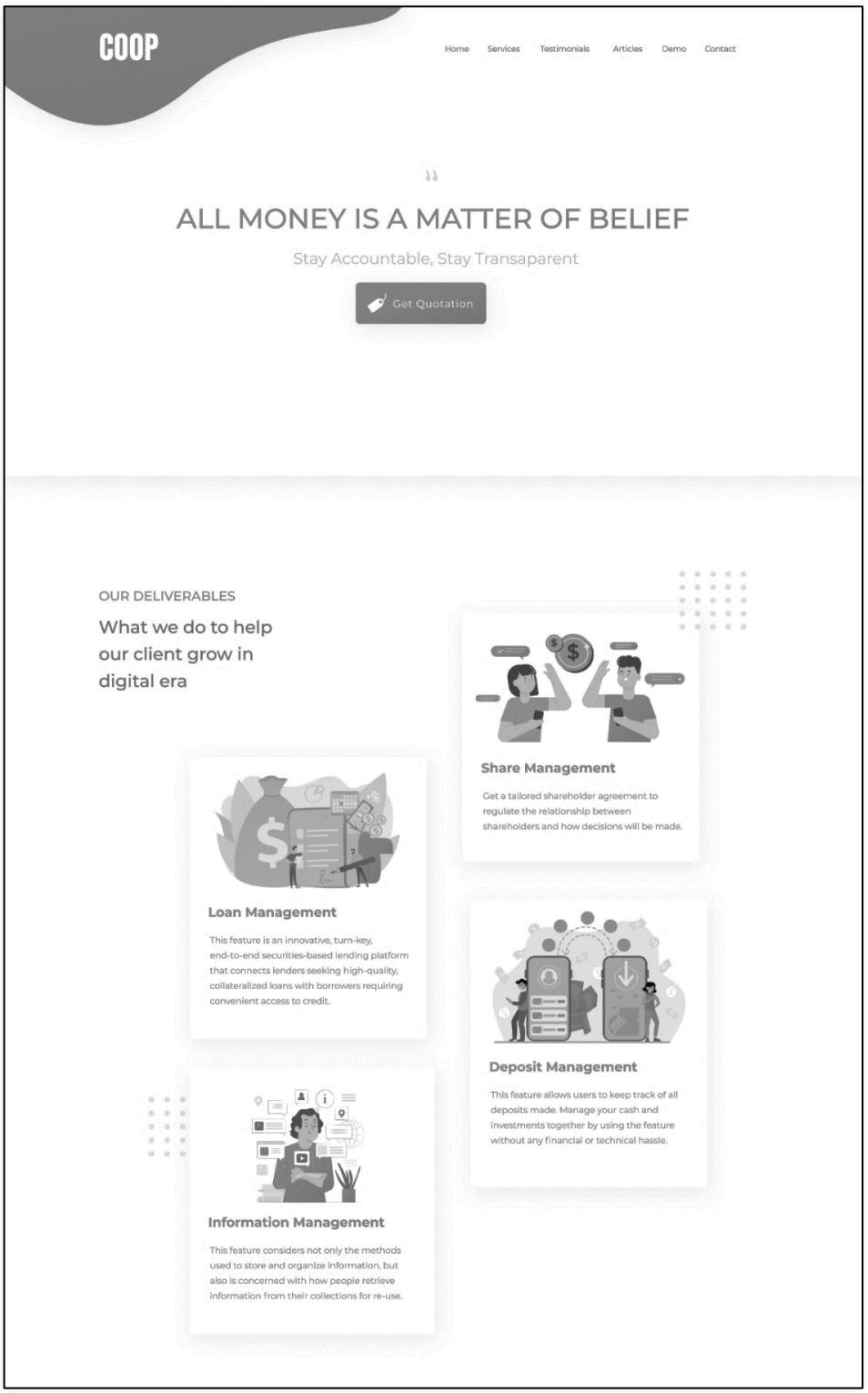


Fig: Promotional website

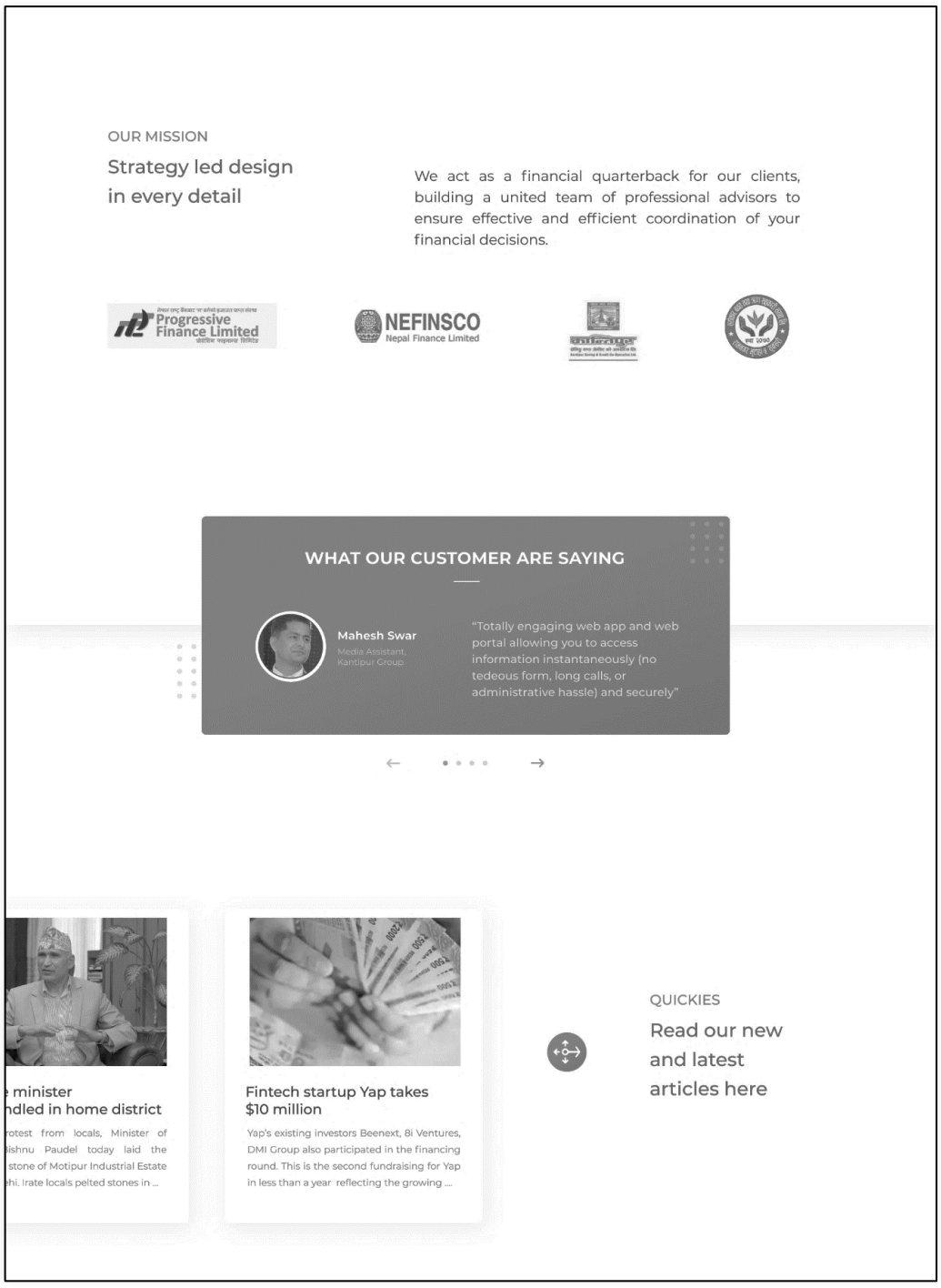


Fig: Promotional website continued

Fig: Promotional website footer section

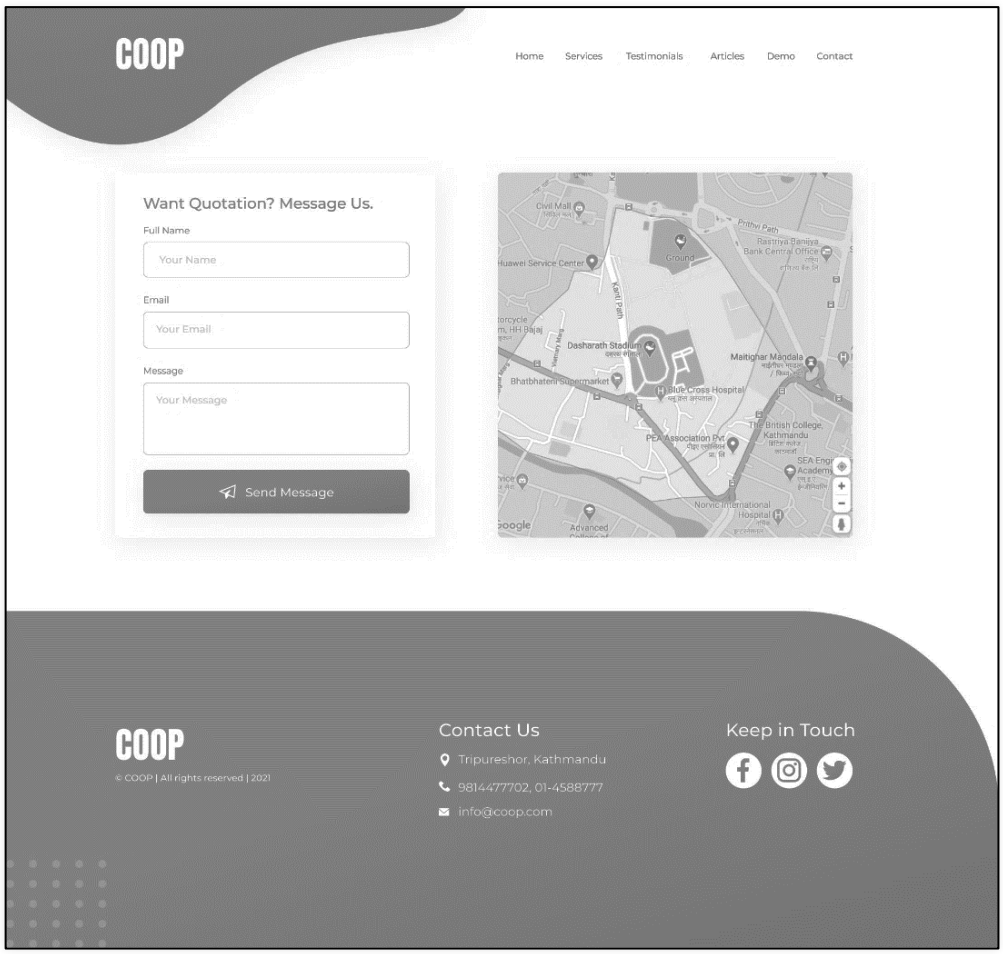


Fig: Promotional website contact page