

# **AN ENTREPRENEUR'S GUIDE TO DEVELOP SOFTWARE FOR BUSINESS**



**QTEC SOLUTION LIMITED**

<b>Table Of Content</b>	<b>3</b>
<b>Chapter 0: Introduction</b>	<b>4</b>
<b>Chapter 1: Check PMF(Product Market Fit) &amp; Gathering requirements. (Image)</b>	<b>8</b>
1.1 What is product-market fit?	8
1.2 What are gathering requirements?	9
1.3 Why is product-market fit important?	11
1.4 What are the requirements gathering in project management?	12
1.5 Five Steps to Achieve product-market Fit	14
1.6 Requirements gathering techniques	15
<b>Chapter 2: Freelancer Or Software Company or Inhouse Team? (bespoke vs OTS)</b>	<b>17</b>
2.1 Who Is a Better Contractor?	17
2.2 Freelancer vs. Software Development Company: PROS and CONS	18
2.3 Helping Businesses to Achieve Their Goals	20
2.4 How much does it cost to hire a software development company? (Image)	20
2.5 How much does it cost to hire a Freelancer? (Image)	21
2.6 What is team augmentation? (Image)	21
2.7 What are the benefits of team augmentation? (Image)	22
2.8 Hiring someone for short-term or long-term? (Image)	23
2.9 What to Choose?	24
<b>Chapter 3: How to Pick the right software company for Your Apps or Website.</b>	<b>25</b>
3.1 How to Choose the Right Software Vendor for Your Business	25
3.2 Why is software vendor evaluation important?	26
3.3 How to choose the right software vendor	26
3.4 8 Things to Consider When Choosing a Software Vendor	28
3.5 The Importance of Software Evaluation	29
<b>Chapter 4: Manage with your In-House Team &amp; Freelancers</b>	<b>30</b>
4.1 How to Manage a software development team	30
4.2 Managing software development teams without having a technical background	32
4.3 How to successfully manage a remote software development team	33
4..4 How To Effectively Manage a Team of Freelancers	35
4.5 Why you should work with freelancers	36
4.6 The difference between managing freelancers vs. in-house employees	37
4.7 What's next?	39
<b>Chapter 5: Tech Maintenance and Support</b>	<b>40</b>
5.1 What is Software Maintenance & Software Support	40
5.2 Software Maintenance: What it Includes, Benefits, Cost, and Process	41
5.3 Software Support: What it Includes, Benefits, Cost, and Process	42
5.4 Software Maintenance and Support Setup Plan	44
5.5 In-house software support & maintenance vs Outsourced software support & maintenance	46
5.6 Are software maintenance and support essential?	47
Conclusion	48

# **Chapter 0:** Introduction

## **Why this book?**

In today's digital age, software has become a critical component of most businesses, regardless of their size or industry. However, many entrepreneurs face significant challenges when it comes to developing software. They may not have the technical expertise, resources, or budget to develop software in-house, and choosing the right software company or freelancer can be a daunting task.

This book aims to address these challenges by providing entrepreneurs with practical guidance on how to develop software for their businesses. It covers every aspect of the development process, from checking the product-market fit and gathering requirements to team management, tech maintenance, and support.

By following the steps outlined in this book, entrepreneurs can save time, money, and resources while ensuring that their software meets the needs of their businesses and their customers. The book is an essential resource for anyone looking to develop software for their business, whether they are a first-time entrepreneur or an experienced business owner.

## **Why is it important?**

"An Entrepreneur's Guide to Develop Software for Business" is important because it provides entrepreneurs with the knowledge and practical guidance they need to develop software for their businesses successfully. It helps entrepreneurs overcome the challenges they may face during the development process and ensures that their software meets the needs of their businesses and their customers.

## **How is it going to help?**

"An Entrepreneur's Guide to Develop Software for Business" is a book that is designed to help entrepreneurs who want to develop software for their business but may not have the technical expertise or resources to do so. The book provides a comprehensive guide that covers every aspect of the software development process, from checking the product-market fit and gathering requirements to team management, tech maintenance, and support.

One of the primary benefits of this book is that it provides entrepreneurs with practical advice and guidance that they can apply in their businesses. By following the steps outlined in the book, entrepreneurs can save time, money, and resources, and avoid common mistakes that can occur during the development process. Additionally, the book provides guidance on how to choose the right development partner, whether it be a freelancer or software company, and how to manage the development team effectively.

The book is written in a friendly and accessible style, making it easy for entrepreneurs to understand and apply the concepts presented. The author uses real-life examples and case studies to illustrate the principles and strategies presented in the book, making it relatable and engaging for readers.

## **Who can benefit from this book?**

"An Entrepreneur's Guide to Develop Software for Business" can benefit a wide range of people, including Entrepreneurs, Startup founders, Small business owners, Project managers, Software Developers. This book can be a valuable resource for anyone involved in software development, from entrepreneurs to project managers to software developers. It provides practical guidance and advice that can help ensure the success of software development projects, whether they are for a small business or a large corporation.

## **Who are we?**

Qtec Solution Limited was founded at the end of 2019 with the aim to engineer robust technology services for super-growing businesses founded by visionaries. We served software solutions for over 30 clients in the last two years and completed 50+ Medium and Large projects.

Most of Qtec Solution Limited's team members have very good experience of working with local and international clients on software development projects.

On-Time delivery and alignment with client's requirements is something we are all about.

## **Why did we write this?**

We likely recognized that many entrepreneurs may not have the technical expertise or resources to develop software on their own, and We wanted to provide a resource that would help them navigate the software development process from start to finish.

The book covers a range of topics, including how to check the product-market fit, gather requirements, choose the right development partner, manage the development team, and maintain the software. By providing practical guidance and advice on each of these topics.

We likely aimed to help entrepreneurs develop software that meets the needs of their businesses and their customers while avoiding common mistakes and pitfalls.

## **Message to entrepreneurs**

Developing software for your business can be a daunting task, especially if you don't have a technical background. However, it can also be an incredibly rewarding endeavor that can help you improve your operations, provide a better customer experience, and grow your business.

That's why we wrote "An Entrepreneur's Guide to Develop Software for Business" - to provide you with a practical guide that can help you navigate the software development process from start to finish. Our book covers a range of topics, from checking the product-market fit to managing the development team and maintaining the software, and provides practical guidance and advice on each of these topics.

We understand that developing software can be a complex and challenging process, but we believe that with the right guidance and resources, anyone can do it. We hope that our book will help you develop software that meets the needs of your business and your customers and that it will help you achieve your goals.

# Chapter 1: Check PMF(Product Market Fit) & Gathering requirements.

## 1.1 What is product-market fit?

Product Market Fit (PMF) refers to the stage in a business's development where its product or service is able to effectively meet the needs and demands of a specific target market. PMF (Product Market Fit) is a concept that describes the degree to which a product satisfies the needs of the market it is intended for. In other words, it measures how well a product fits the market it is designed for.

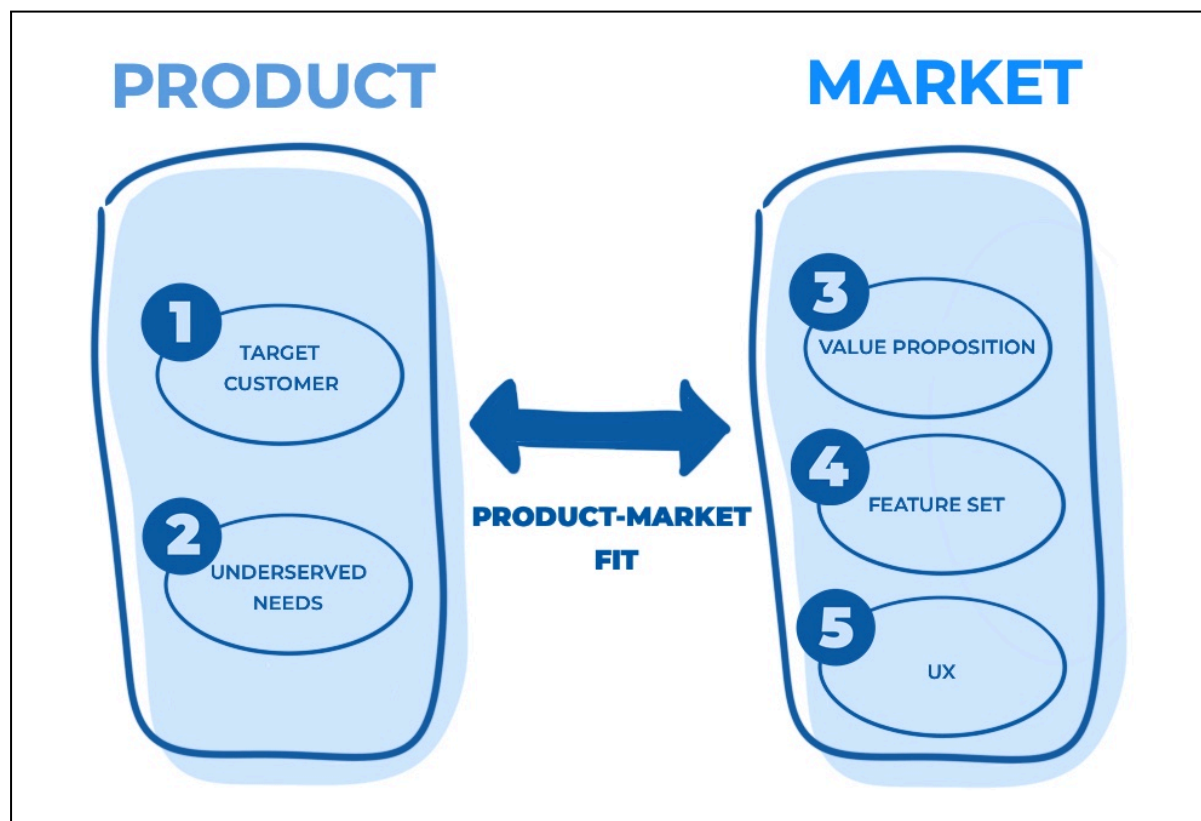


Image Credit: Akita Solutions

A common indication of attaining PMF is when consumers display a willingness to purchase the product, utilize it on a regular basis, and recommend it to others, despite any minor imperfections that may still exist.

To assess PMF, you need to gather data on customer feedback and usage metrics, such as retention rates, user engagement, and customer satisfaction. This data can help you identify patterns and trends in customer behavior, which can help you improve your product and better align it with your target market.

## 1.2 What are gathering requirements?

Gathering requirements is the process of identifying and defining the needs and expectations of your customers, stakeholders, and end-users. This process involves collecting and analyzing information about the features, functionality, and performance of your product or service, as well as any constraints or limitations that may affect its development and delivery.

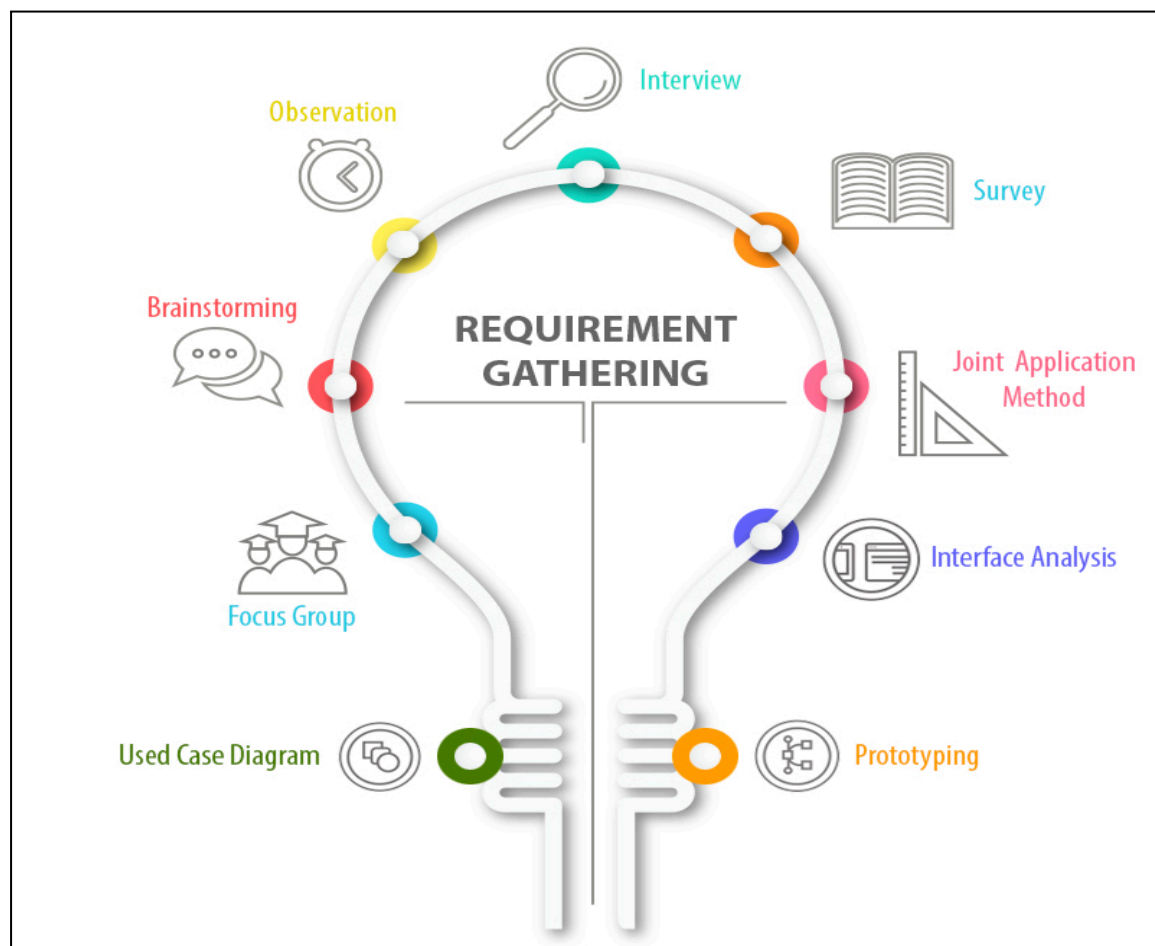


Image credit: Analytics Vidhya

To gather requirements effectively, you can use various techniques, such as surveys, focus groups, interviews, and user testing. You should also involve your stakeholders and end-users throughout the process to ensure their needs and

expectations are fully understood and incorporated into the final product. Additionally, it's important to prioritize the requirements based on their importance and feasibility, so that you can deliver a product that meets the most critical needs of your target market.

It's worth noting that PMF and gathering requirements are interconnected, as gathering requirements helps you understand the market demand and user needs, which are crucial to achieving PMF. By gathering feedback and requirements from users and stakeholders, you can refine your product to better meet their needs and increase your chances of achieving PMF.

### 1.3 Why is product-market fit important?

Product-market fit (PMF) is essential because it determines the success or failure of a product. PMF is the degree to which a product satisfies the market demand for it, and it indicates that the product has found a strong fit with the needs and wants of its target market. Here are a few reasons why achieving PMF is crucial for any product:



### 1.3 Imprtants of product market fit



- **Increased user adoption:** When a product meets the needs and wants of its target market, users are more likely to adopt it, which can lead to increased sales, revenue, and market share.
- **Reduced marketing costs:** A product that has achieved PMF is more likely to generate organic growth and word-of-mouth marketing, reducing the need for expensive marketing campaigns.
- **Competitive advantage:** A product that has achieved PMF has a competitive advantage over similar products that do not meet the needs and wants of the target market.
- **Increased customer satisfaction:** When a product meets the needs and wants of its target market, users are more likely to be satisfied with it, leading to increased loyalty and repeat purchases.

In summary, achieving PMF is crucial for the success of any product, as it helps increase user adoption, reduce marketing costs, provide a competitive advantage, and increase customer satisfaction.

## 1.4 What are the requirements gathering in project management?

Requirements gathering is the process of identifying and documenting the needs and expectations of stakeholders for a project. It is a critical activity in project management as it helps ensure that the project meets its objectives and satisfies the requirements of its stakeholders. The process of collecting requirements often consists of the subsequent stages:



## 1.4 Requirement gathering in project management

- **Identify stakeholders:** The first step in requirements gathering is to identify all the stakeholders involved in the project. Stakeholders can include project sponsors, end-users, customers, and other relevant parties.
- **Define objectives:** The project objectives should be clearly defined to ensure the requirements-gathering process is focused and relevant.
- **Conduct interviews and workshops:** The project team should conduct interviews and workshops with the stakeholders to gather information about their needs, expectations, and concerns. This can include one-on-one interviews, group workshops, and surveys.
- **Analyze data:** The information gathered from the interviews and workshops should be analyzed to identify common themes, patterns, and requirements.

- **Document requirements:** The requirements should be documented in a clear and concise manner, using appropriate tools such as use cases, user stories, and requirement specifications.
- **Validate requirements:** The requirements should be validated with the stakeholders to ensure they are accurate, complete, and meet their needs and expectations.
- **Manage requirements:** The requirements should be managed throughout the project lifecycle to ensure they are met and any changes are documented and communicated to stakeholders.

In summary, requirements gathering is a critical activity in project management that involves identifying and documenting the needs and expectations of stakeholders to ensure that the project meets its objectives and satisfies the requirements of its stakeholders.

### 1.5 Five Steps to Achieve product-market Fit



#### 1.5 Achieve product-market Fit

Achieving product-market fit is a critical step in building a successful product. Here are five steps to achieve product-market fit:

- **To accomplish the product-market fit, the initial step is to pinpoint your intended audience:** identifying the specific group of people who are most likely to have a need for your product or service. The first step in achieving product-market fit is to identify your target market. You need to understand the characteristics of your potential customers, including their demographics, psychographics, and behaviors. This will help you tailor your product to meet their specific needs.
- **Define your value proposition:** Your value proposition is the unique benefit your product provides to your target market. You need to define your value proposition clearly and concisely so that it resonates with your target market and differentiates your product from the competition.
- **Build a Minimum Viable Product (MVP):** A Minimum Viable Product is a version of your product that has just enough features to meet the needs of early adopters. You should focus on building an MVP quickly and efficiently, using feedback from early adopters to refine your product and improve its market fit.
- **Gather feedback and iterate:** Once you have launched your MVP, you need to gather feedback from early adopters and use it to refine your product. You should be willing to iterate quickly and make changes to your product based on the feedback you receive.
- **Scale and grow:** Once you have achieved product-market fit, you can focus on scaling and growing your product. You should continue to gather feedback from your customers and use it to improve your product and meet the evolving needs of your target market.

In summary, achieving product-market fit requires you to identify your target market, define your value proposition, build an MVP, gather feedback and iterate, and scale and grow your product. By following these five steps, you can increase the likelihood of building a successful product that meets the needs of your target market.

## 1.6 Requirements gathering techniques

Requirements-gathering techniques are methods used to collect information and requirements from stakeholders in a project. Listed below are a few frequently employed methods for collecting requirements:

- **Interviews:** One-on-one interviews with stakeholders can help to gather in-depth information about their needs and requirements. Interviews are particularly useful when gathering information from key stakeholders, such as project sponsors, subject matter experts, and end-users.
- **Workshops:** Workshops are group sessions that allow stakeholders to collaborate and share their ideas, opinions, and requirements. Workshops can be particularly useful when gathering requirements from a diverse group of stakeholders or when brainstorming new ideas.
- **Surveys:** Surveys are a cost-effective and efficient way to gather information from a large number of stakeholders. Surveys can be used to gather information about stakeholders' opinions, preferences, and requirements.
- **Focus groups:** Focus groups are group discussion led by a facilitator that allows stakeholders to share their opinions and feedback on a particular topic. Focus groups can be particularly useful when testing new product concepts or designs.
- **Observation:** Observing stakeholders as they work can help to identify their needs and requirements. Observation can be particularly useful when gathering requirements for software or systems that are currently in use.
- **Prototyping:** Building a prototype of a product can help to gather feedback and requirements from stakeholders. Prototyping can be particularly useful when developing user interfaces or testing new product concepts.
- **Document analysis:** Analyzing existing documents, such as reports, specifications, and standards, can help to identify requirements and understand the context of the project.

In summary, there are many different requirements-gathering techniques, each with its strengths and weaknesses. By using a combination of techniques, project

managers can gather the information they need to build successful products that meet the needs of their stakeholders.

### References

- <https://asana.com/resources/requirements-gathering>
- <https://ahrefs.com/blog/product-market-fit/>

## **Chapter 2:** Freelancer Or Software Company or Inhouse Team? (bespoke vs OTS)

### **2.1 Who Is a Better Contractor?**

The decision of whether to hire a freelancer or a software development company as a contractor depends on several factors, such as project scope, budget, timeline, and desired level of expertise. Both freelancers and software development companies have their own advantages and disadvantages, and the right choice will depend on the specific needs of the project.

You are probably wondering whether to hire a freelancer or seek assistance from a software development company. The decision you make now will influence the whole software development process and impact the future success of your project.

Freelancers are typically individuals who work independently and offer their services to clients on a project basis. They may have specialized skills in a particular area of software development, and can often offer competitive rates due to their lower overhead costs. Freelancers can be a good choice for smaller projects or projects with a limited budget, as they may be more flexible and able to adapt to changing project requirements.

Software development companies, on the other hand, are typically larger organizations that employ teams of developers with a range of skills and expertise. They may offer a more comprehensive set of services, including project management, quality assurance, and ongoing support. They may also have more established processes and procedures in place to ensure consistent quality and timely delivery of projects. Software development companies can be a good choice for larger or more complex projects, as they may have the resources and expertise to handle the scope and complexity of the project.

Ultimately, the decision of whether to hire a freelancer or a software development company will depend on the specific needs of the project, as well as factors such as budget, timeline, and desired level of expertise. It's important to carefully evaluate potential contractors and choose one that has the skills and experience necessary to deliver a successful project.

## **2.2 Freelancer vs. Software Development Company: PROS and CONS**



### 2.2 Freelancer vs. Software Development Company

Both freelancers and software development companies have their own pros and cons when it comes to contracting for software development projects. Below are some of the pros and cons associated with each option:

#### **Freelancer Pros:**

- **Lower cost:** Freelancers often have lower rates than software development companies due to their lower overhead costs.
- **Flexible:** Freelancers can often work on a more flexible schedule and can be more adaptable to changes in project requirements.
- **Specialized skills:** Freelancers often have specialized skills in a particular area of software development.
- **Personal touch:** Working with a freelancer can feel more personal as there is a direct line of communication between the client and the contractor.

#### **Freelancer Cons:**

- **Limited resources:** Freelancers typically have limited resources and may not be able to handle larger or more complex projects.
- **Lack of support:** Freelancers may not offer the same level of support as software development companies, particularly after the project is completed.
- **Unreliable:** Freelancers may not always be reliable, particularly if they have multiple projects or are unavailable for personal reasons.

#### **Software Development Company Pros:**

- **Established processes:** Software development companies often have established processes and procedures in place to ensure consistent quality and timely delivery of projects.
- **The broad range of skills:** Software development companies often have teams of developers with a range of skills and expertise.
- **More resources:** Software development companies typically have more resources and can handle larger or more complex projects.
- **Ongoing support:** Software development companies often offer ongoing support after the project is completed.

#### **Software Development Company Cons:**



- **Higher cost:** Software development companies often have higher rates than freelancers due to their higher overhead costs.
- **Less flexible:** Software development companies may be less flexible when it comes to changes in project requirements or working on a specific schedule.
- **Less personal touch:** Working with a software development company can feel less personal as there may be more layers of communication between the client and the contractor.

## 2.3 Helping Businesses to Achieve Their Goals



2.3 Helping Businesses to Achieve Their Goals

The purpose of either a software development company or a freelancer is to help businesses (or individuals) achieve their goals and get the most out of their websites/web solutions. Now, the difference between a freelancer and an entire software development company is quite straightforward. A team vs. a member. Employees that have expertise in several specializations or someone specialized in a particular field.

## **2.4 How much does it cost to hire a software development company?**



2.4 How much does it cost to hire a software development company

The cost of hiring a software development company can vary widely depending on several factors such as the complexity of the project, the location of the development team, the level of experience of the developers, and the duration of the project.

Generally, software development companies charge either on an hourly basis or a fixed project basis. The hourly rate can range from \$50 to \$300 or more depending on the location and the experience of the development team.

For a fixed project basis, the cost can range from a few thousand dollars to several hundred thousand dollars depending on the complexity and the scope of the project.

It's important to note that the cost is just one of the factors to consider when hiring a software development company. It's also important to consider their experience, expertise, and reputation in the industry, as well as their ability to meet project timelines and deliver high-quality work. It's a good idea to research and compare different software development companies to find the one that best fits your needs and budget.

## **2.5 How much does it cost to hire a Freelancer?**

The cost of hiring a freelancer can vary widely depending on several factors such as the freelancer's level of experience, the complexity of the project, and the location of the freelancer.

Freelancers typically charge either on an hourly basis or a fixed project basis. The hourly rate can range from \$20 to \$200 or more depending on the freelancer's experience and the type of work required.

When working on a project with a fixed budget, the expenses may vary from a few hundred dollars to several thousand dollars, depending on the intricacy and extent of the project..

It's important to note that the cost is just one of the factors to consider when hiring a freelancer. It's also important to consider their experience, skills, and reputation in the industry, as well as their ability to meet project timelines and

deliver high-quality work. It's a good idea to research and compare different freelancers to find the one that best fits your needs and budget.

## 2.6 What is team augmentation?

# TEAM AUGMENTATION



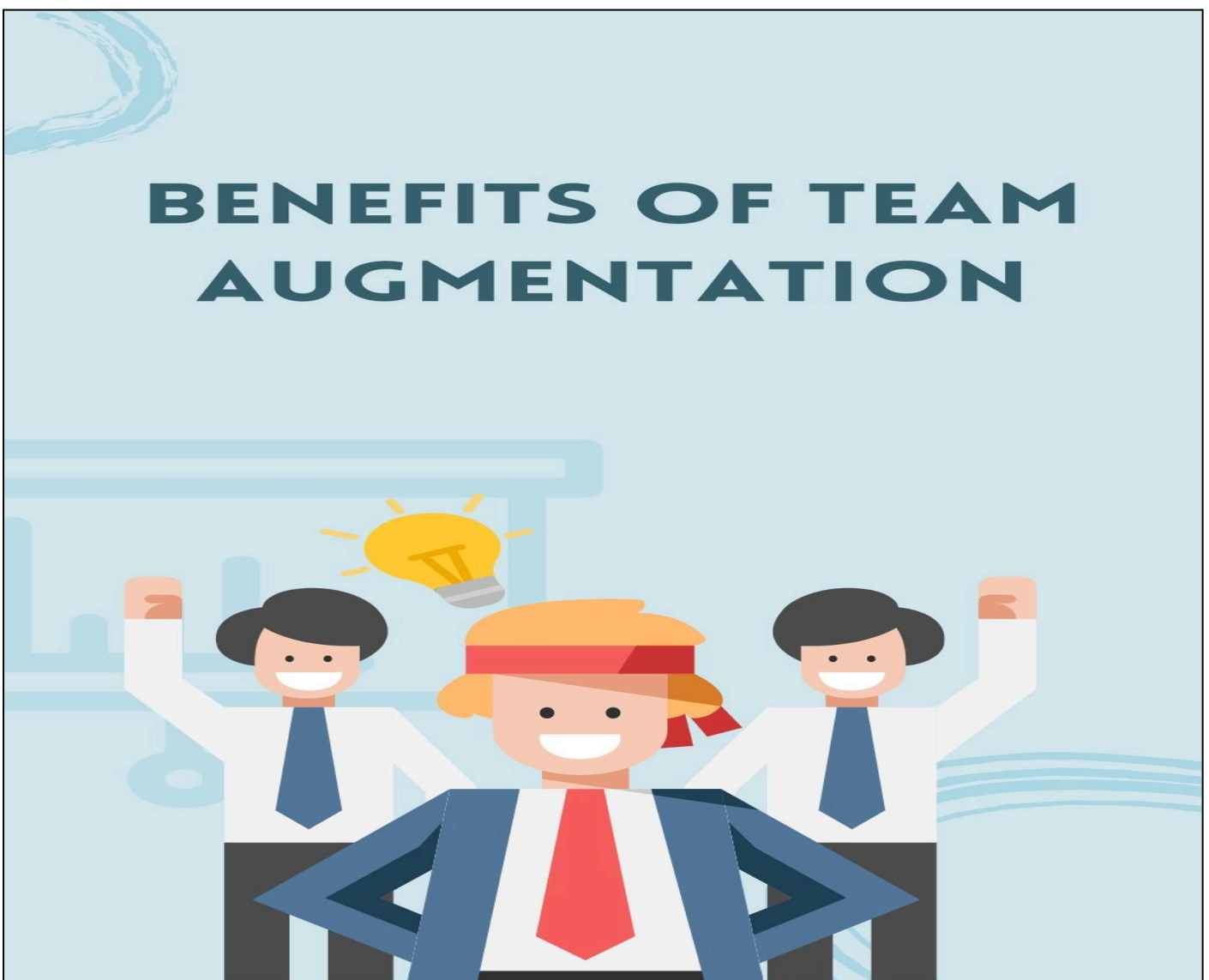
## 2.6 Team augmentation

Team augmentation is the process of hiring external talent to work alongside your existing team members. This approach can help businesses to quickly scale their teams, access specialized skills and expertise, and complete projects faster and more efficiently.

Team augmentation can be particularly useful in industries where the demand for skilled workers is high, but the supply is limited. By tapping into a broader pool of talent, businesses can find the specialized skills they need to complete complex projects, without having to invest in full-time employees.

Overall, team augmentation can be a cost-effective, flexible, and efficient way to complement your existing team's skills and expertise, and to take on more ambitious projects with greater confidence.

## **2.7 What are the benefits of team augmentation?**



## **2.7 What are the benefits of team augmentation?**

The practice of team expansion involves the recruitment of additional external professionals to collaborate with your current team members. Here are some benefits of team augmentation:

- **Access to specialized skills:** Augmenting your team can provide access to specialized skills that may not be available within your existing team. This can help you to complete projects faster and to a higher standard.
- **Flexibility:** Augmenting your team allows you to scale up or down as needed. This can be especially useful if you have a project with a tight deadline, as you can quickly bring in additional resources to help you meet your goals.
- **Cost-effective:** Augmenting your team can be more cost-effective than hiring full-time employees, as you only pay for the work completed and do not have to provide benefits or other perks.
- **Reduced risk:** Augmenting your team can help to reduce risk, as you do not have to commit to a long-term contract or make a significant investment in hiring someone full-time.
- **Faster time to market:** By augmenting your team, you can bring in additional resources to help you get your product or service to market faster. This can help you to stay ahead of your competitors and to capture market share more quickly.

Overall, team augmentation can be a great way to improve your team's productivity, reduce costs, and accelerate your time to market. By bringing in external talent, you can complement your existing team's skills and expertise, and take on more ambitious projects with greater confidence.

## 2.8 Hiring someone for short-term or long-term?

Whether to hire someone for short-term or long-term depends on the specific needs and goals of the business. Below are a few factors to take into account while contemplating this choice:

### Short-term hiring:

- **Flexibility:** Short-term hiring provides flexibility, as you can hire someone for a specific project or task, and then end the contract once the work is completed.
- **Cost-effective:** Hiring someone for short term can be more cost-effective than hiring a full-time employee, as you only pay for the work completed and do not have to provide benefits or other perks.
- **Specialized skills:** If you need someone with specialized skills for a specific project or task, short-term hiring can be a good option.
- **Risk management:** Short-term hiring can be a good way to manage risk, as you do not have to commit to a long-term contract or make a significant investment in hiring someone full-time.

### Long-term hiring:

- **Stability:** Long-term hiring provides stability, as you have a consistent employee who is committed to the success of your business.
- **Cultural fit:** Long-term employees have more time to understand and align with your company culture, which can be important for long-term success.
- **Cost-effective:** Long-term employees can be more cost-effective in the long run, as they require less training and can become more productive over time.
- **Growth:** If you have long-term goals for your business, hiring someone full-time can be a good way to support growth and build a strong team.

Ultimately, the decision to hire someone for the short-term or long-term depends on the specific needs and goals of your business. Consider the factors above, as well as the specific project or role you need to fill, and choose the option that best fits your needs.

## 2.9 What to Choose?

There is no straightforward response to whether or not to hire a freelancer vs. hiring a software development company. Selecting the optimal option can be challenging as numerous factors can impact the development processes. Although freelancers can provide more flexibility and cost savings, outsourcing software development companies generally surpass them in terms of expertise, utilization of modern technology stacks, and development turnaround times.

### References

- <https://medium.com/@zfort/freelancer-vs-software-development-company-who-is-a-better-contractor-9531d219ce9a/>
- <https://tagdiv.com/freelancer-vs-software-development-company-pros-and-cons>
- <https://www.lazorpoint.com/insights/prosandcons/>
- <https://www.trentonsystems.com/blog/in-house-vs-outsourced-technical-support/>
- <https://www.one-beyond.com/team-augmentation-secret-supercharged-business/>



## **Chapter 3:** How to Pick the right software company for Your Apps or Website.

### **3.1 How to Choose the Right Software Vendor for Your Business**



### **3.1 How to Choose the Right Software Vendor for Your Business**

Choosing the right software vendor is critical to the success of your business. Navigating the marketing and sales language surrounding a new SaaS tool can be challenging. The decision has the potential to make or break your enterprise and the workflow of various teams. Implementing the wrong software, or even deploying good software the wrong way, can set your employees up for failure before they even get started.

When choosing software vendor partners, it is crucial to ensure a good fit for both parties. Conducting a software vendor evaluation is essential to determine compatibility.

### 3.2 Why is software vendor evaluation important?

Software vendor evaluation is important because it allows businesses to make informed decisions when selecting software vendors. Evaluating software vendors ensures that the software chosen meets the needs of the business and is reliable, secure, and scalable. By evaluating potential vendors based on criteria such as reputation, experience, expertise, security, scalability, and pricing, businesses can select the best vendor to provide software that meets their requirements.

This ensures that the business can operate efficiently and effectively, while minimizing the risk of software-related issues such as downtime, security breaches, or insufficient functionality. Ultimately, software vendor evaluation is crucial to ensuring that the software chosen is a good fit for the business and provides value in the long term.

### 3.3 How to choose the right software vendor

Choosing the right software vendor is an important decision for any business, as it can have a significant impact on productivity, efficiency, and overall success. Here are some steps to follow when choosing a software vendor:

- **Define your requirements:** Before you begin your search for a software vendor, you need to define your requirements. Identify the specific features and functionality that your business needs in the software, as well as any integration requirements with existing systems.
- **Research potential vendors:** Once you have a clear idea of your requirements, research potential vendors who specialize in the type of

software you need. Look for vendors with a proven track record of success and experience in your industry.

- **Evaluate vendor reputation:** Check the vendor's reputation by reading reviews from current and former customers, checking references, and reviewing case studies or testimonials. This will give you an idea of the vendor's level of expertise, customer service, and overall reliability.
- **Assess vendor expertise:** Assess the vendor's expertise in the area of software development you require. Find out about their development process, quality control, and bug-fixing procedures.
- **Consider scalability:** Consider the vendor's ability to support your business as it grows. Make sure that the software is scalable and can accommodate increased demand, without requiring significant additional investment in new software or services.
- **Evaluate vendor security:** Evaluate the vendor's security and data protection practices. Check if the vendor is compliant with relevant standards and regulations, and if they provide robust data backup and recovery solutions.
- **Review pricing and support:** Review the vendor's pricing model and support options. Determine if their pricing is transparent and affordable, and whether they offer sufficient support, training, and ongoing maintenance.
- **Test the software:** Test the software before you make a final decision. Most vendors will offer a free trial or demo of their software, which will allow you to see if the software meets your requirements.

By following these steps, you can choose the right software vendor for your business, which will provide software that meets your specific needs, is reliable, secure, and scalable, and will help you achieve your business goals.

### 3.4 8 Things to Consider When Choosing a Software Vendor

When choosing a software vendor, there are several important factors to consider. Here are eight things to keep in mind:

- **Industry expertise:** Look for a software vendor with experience in your industry. They should understand your business needs and be able to offer solutions that are tailored to your specific requirements.
- **Reputation:** Consider the vendor's reputation in the industry. Look for reviews, testimonials, and references from current and former customers to get a sense of their reliability, quality of work, and level of customer service.
- **Track record:** Review the vendor's track record. Have they delivered successful projects in the past? What is their success rate?
- **Quality of work:** Assess the vendor's quality of work. Do they follow best practices for software development? Do they have a rigorous testing process?
- **Security:** Evaluate the vendor's security practices. Do they have appropriate security measures in place to protect your data and systems? Are they compliant with relevant standards and regulations?
- **Scalability:** Determine if the software can grow with your business. Look for a vendor that offers solutions that can scale to meet your changing needs.
- **Support:** Consider the level of support provided by the vendor. Do they offer ongoing maintenance and technical support? Are they available when you need them?
- **Pricing:** Review the vendor's pricing model. Is it transparent and easy to understand? Are there any hidden costs or fees?

By considering these factors when choosing a software vendor, you can select a partner that will deliver quality software that meets your needs and helps your business succeed.

### 3.5 The Importance of Software Evaluation

Software evaluation is an essential process for businesses that are looking to implement new software solutions or upgrade their existing ones. Here are some reasons why software evaluation is important:

- **Identifying the right software:** Software evaluation allows businesses to identify the right software for their specific needs. By assessing different

software options and evaluating them against their requirements, businesses can select software that meets their unique needs and goals.

- **Ensuring compatibility:** Software evaluation helps businesses ensure that the software they select is compatible with their existing systems and infrastructure. This approach assists in reducing interruptions and enhancing effectiveness.
- **Reducing risks:** Evaluating software before implementation can help reduce the risks associated with software failure, downtime, and security breaches. By selecting software that has been thoroughly tested and vetted, businesses can reduce the likelihood of experiencing these issues.
- **Maximizing ROI:** By selecting the right software, businesses can maximize their return on investment. Software that is tailored to their needs and goals can help increase productivity, efficiency, and profitability.
- **Ensuring user adoption:** Software evaluation helps ensure that the software selected is easy to use and adopted by end-users. This can help increase user satisfaction and reduce the risk of resistance to change.
- **Future-proofing:** By evaluating software against their future needs, businesses can future-proof their investment. This helps ensure that the software will continue to meet their needs as their business evolves and grows.

In conclusion, software evaluation is an essential process that helps businesses select software that meets their unique needs and goals, reduces risks, maximizes ROI, ensures user adoption, and future-proofs their investment.

## References

- <https://track.g2.com/resources/software-vendor-evaluation>
- <https://start-up.house/en/blog/articles/choosing-a-software-vendor>
- <https://www.cio.com/article/289166/how-to-choose-the-right-software-vendor.html>

# Chapter 4: Manage with your In-House Team & Freelancers

## 4.1 How to Manage a software development team



## 4.1 How to Manage a software development team

Managing a software development team is a complex task that requires a combination of technical expertise, leadership skills, and effective

communication. To manage a software development team effectively, you should:

- **Set clear goals and expectations:** When you establish clear goals and expectations, your team will have a shared understanding of what they are working towards and what is expected of them. This will help your team stay focused and motivated.
- **Create a positive work environment:** A positive work environment can boost morale and productivity. This includes encouraging your team to work collaboratively, recognizing their accomplishments, and creating an open and supportive workplace culture.
- **Foster open communication:** Encourage your team to communicate openly with each other and with you. This can be achieved through regular team meetings, one-on-one conversations, and even team-building activities. By fostering open communication, you can help ensure that everyone is on the same page and working towards the same goal.
- **Set realistic timelines and priorities:** Setting realistic timelines and priorities can help your team stay focused and motivated. Work with your team to identify the most important tasks and create a timeline that is achievable. Make sure to build in some flexibility to account for unforeseen challenges or obstacles.
- **Provide resources and support:** Providing your team with the resources they need to do their jobs effectively is essential for success. This may include access to the necessary tools and technology, training, and support from other team members or mentors.
- **Encourage personal and professional growth:** Encouraging your team to learn and grow both personally and professionally is important for their long-term success. Offer opportunities for training and development, and recognize and reward their achievements.
- **Monitor progress and provide feedback:** Regularly monitoring your team's progress and providing feedback on their work is critical for keeping everyone on track and ensuring that the project stays on course. Be sure to recognize when your team is doing a good job, and offer constructive feedback when there is room for improvement.



By following these tips, you can help your software development team be successful and achieve their goals. Remember to lead by example, communicate clearly and openly, and provide support and resources to help your team do their best work. With the right approach, managing a software development team can be a rewarding experience for everyone involved.

## 4.2 Managing software development teams without having a technical background

Managing a software development team without a technical background can be a daunting task, but with the right approach, it is possible to lead a successful team. Here are some additional tips to help you manage a software development team without a technical background:

- **Focus on the business value:** As a non-technical manager, it's essential to focus on the business value of your software development project. Understand what problem the software is trying to solve, and how it will benefit your organization. This will help you communicate the project's goals and priorities to your team more effectively.
- **Build relationships with technical team members:** Building a strong relationship with technical team members is crucial for success. Make an effort to get to know your team members and their technical expertise. By understanding their skills, you'll be better equipped to identify any gaps in your knowledge and make informed decisions.
- **Develop a basic understanding of software development:** While you may not need to be an expert in software development, having a basic understanding of the process and the technology involved can be beneficial. Consider taking a course or attending a workshop to learn more about software development.
- **Trust your team:** As a non-technical manager, it's important to trust your team's technical expertise. Allow your team members to make technical decisions and provide guidance and support as needed.
- **Creating unambiguous communication channels is essential:** Efficient communication is vital for the triumph of every project. Establish clear communication channels with your team, and encourage open and honest communication.



- **Use metrics to measure progress:** Use metrics such as the number of completed tasks, time taken to complete tasks, and software quality to measure the progress of your project. This will help you identify any areas that need improvement and keep the project on track.
- **Hire a technical advisor:** Consider hiring a technical advisor or consultant to provide guidance and advice on technical matters. This can be particularly useful if you're working on a complex software development project.

Managing a software development team without a technical background is challenging, but it is possible with the right approach. By building relationships with your team members, focusing on the business value, and establishing clear communication channels, you can lead a successful software development project. Remember to trust your team's technical expertise, and don't be afraid to ask for help or guidance when needed.

## 4.3 How to successfully manage a remote software development team

Managing a remote software development team can present unique challenges, such as communication barriers and a lack of face-to-face interaction. However, with the right approach and tools, it is possible to manage a successful remote team. Here are some additional tips to help you successfully manage a remote software development team:

- **Define roles and responsibilities:** It's crucial to define the roles and responsibilities of each team member to ensure everyone is working towards the same goal. This will help minimize confusion and ensure everyone is contributing to the project in a meaningful way.
- **Set up a strong onboarding process:** An effective onboarding process is essential for setting up a new remote team member for success. This includes providing clear documentation, training materials, and access to the necessary tools and systems.
- **Foster a culture of trust:** Trust is essential when working with a remote team. Encourage open and honest communication, and make sure team members feel comfortable asking for help or guidance when needed. Trust

also involves allowing team members to work independently without micromanaging.

- **Be mindful of time zones:** If your team is spread across different time zones, it's important to be mindful of this when scheduling meetings or setting deadlines. Consider using a time zone converter to ensure everyone is on the same page.
- **Ensure reliable internet connectivity:** Reliable internet connectivity is essential for a remote team to work efficiently. Encourage team members to have a backup plan in case of internet or power outages.
- **Encourage social interaction:** Social interaction is critical for building team cohesion, even in a remote environment. Consider scheduling regular virtual team-building activities, such as virtual game nights or coffee chats.
- **Use project management tools:** There are many project management tools available for remote teams, such as Asana or Jira. Using these tools can make it easier to keep everyone on track and ensure everyone is on the same page.
- **Provide ongoing feedback and recognition:** Providing regular feedback and recognition to team members can help keep them motivated and engaged. This includes providing constructive feedback on areas for improvement as well as recognizing their successes.

Managing a remote software development team requires some additional considerations, but with the right approach, it can be a successful and rewarding experience. By defining roles and responsibilities, setting up a strong onboarding process, fostering a culture of trust, being mindful of time zones, encouraging social interaction, using project management tools, and providing ongoing feedback and recognition, you can lead a successful remote team.

## 4..4 How To Effectively Manage a Team of Freelancers

Managing a team of freelancers can be a great way to tap into specialized skills and expertise that may not be available in-house. However, it also requires a different approach to management than managing a team of in-house

employees. Here are some tips to help you effectively manage a team of freelancers:

- **Clearly Define Project Goals:** Start by making sure everyone on the team understands the project's goals and their role in achieving them. Set clear expectations and deadlines to ensure everyone is on the same page.
- **Establish Communication Channels:** Communication is key when managing a team of freelancers. Establish clear channels, such as email or messaging apps, and set expectations for response times.
- **Provide Regular Feedback:** Freelancers need feedback to help them improve and understand their role. Provide regular feedback, and take the time to build a positive relationship with them.
- **Set Clear Payment Terms:** Discuss payment terms up front and include them in the contract. Clearly define the payment schedule and tie payments to specific milestones or deliverables.
- **Trust and Delegate:** Freelancers are often hired for their specialized skills, so trust their expertise and delegate tasks accordingly. Give freelancers the autonomy to work independently, but be available to provide guidance and support when needed.
- **Build a Strong Working Relationship:** Building a strong working relationship with freelancers is important, even though they work remotely. Take the time to get to know them personally and establish trust. Doing so can foster a favorable work atmosphere and enhance teamwork.

By following these tips, you can effectively manage a team of freelancers and ensure that everyone is working together towards the project's goals. Remember that freelancers are independent contractors, but they still need guidance and support to be successful. With the right approach, you can build a productive and positive relationship with your team of freelancers.

## 4.5 Why you should work with freelancers

Working with freelancers can provide many benefits for businesses, from cost savings to access to specialized skills and expertise. Here are some reasons why you should consider working with freelancers:

- **Cost Savings:** Freelancers often charge lower rates than in-house employees, as they don't require the same level of overhead costs, such as benefits, office space, and equipment.
- **Flexibility:** Freelancers offer flexibility in terms of project duration and workload. You can hire them for a specific project or task, without the commitment of a long-term employment contract.
- **Access to Specialized Skills:** Freelancers offer access to specialized skills and expertise that may not be available in-house. They often have experience in specific industries or areas of expertise and can bring fresh perspectives and ideas to your projects.
- **Scalability:** Freelancers can help businesses scale up or down quickly, depending on project demands. This can help you save time and resources, as you don't have to worry about recruiting and training new employees.
- **Increased Productivity:** Freelancers often work remotely and are not subject to the same office distractions and interruptions as in-house employees. This can help increase productivity and efficiency.
- **Innovation:** Freelancers often work with multiple clients, which exposes them to different industries and approaches. By doing so, you can boost productivity and streamline efficiency.

By working with freelancers, businesses can tap into a wide range of benefits that can help them save time, and resources, and improve project outcomes.

However, it's important to approach freelancers as valuable members of the team and establish clear communication and expectations. By fostering a positive working relationship with freelancers, you can maximize their impact on your business.

## 4.6 The difference between managing freelancers vs. in-house employees

Managing freelancers and in-house employees can present different challenges and require different approaches. Here are some key differences to consider:

- **Communication:** Freelancers may have different communication preferences or time zone differences, which can make communication more challenging than with in-house employees. It's important to establish clear communication expectations and channels from the start.
- **Availability:** Freelancers typically have more flexible schedules and may work on multiple projects simultaneously. In-house employees generally have set schedules and are dedicated solely to your company.
- **Commitment:** Freelancers are not bound to your company in the same way that in-house employees are. They may have multiple clients or other commitments, which can affect their availability and dedication to your project.
- **Expertise:** Freelancers are often hired for their specialized skills and expertise in a specific area, while in-house employees may have a wider range of skills and experience.
- **Culture:** In-house employees are a part of the company culture and have a deeper understanding of the company's values and goals. Freelancers may not have the same level of familiarity with the company culture and may require more guidance in this area.
- **Management:** Freelancers may require less management than in-house employees, as they are typically more self-sufficient and work independently. In-house employees may require more guidance and supervision to ensure they are meeting company standards and goals.
- **Cost:** In-house employees typically have a higher cost than freelancers, as they require benefits, office space, and other resources. Freelancers are generally paid on a project-by-project basis and do not require the same level of resources.

Overall, managing freelancers requires a different approach than managing in-house employees. Freelancers require clear communication, established

expectations, and management, but are often hired for their specialized expertise and lower cost. In-house employees require a deeper understanding of the company culture, ongoing management, and supervision, and have a higher cost but may offer more versatility and dedication to the company's goals. The choice between hiring freelancers and in-house employees ultimately depends on the specific needs and goals of the project or company.

## **4.7 What's next?**

Now that you understand the benefits of working with freelancers and in-house software development teams, the next step is to evaluate which option is best for your business based on your specific needs and goals.

Consider factors such as project scope, budget, timeline, and the skills and expertise required to successfully complete the project. If the project requires specialized skills or expertise that is not available in-house, then working with a freelancer may be the best option. On the other hand, if the project is ongoing or requires consistent collaboration, an in-house team may be a better fit.

Once you have determined which option is best, it's important to establish clear communication and expectations with your team. This includes setting project goals, deadlines, and milestones, as well as establishing communication channels and payment terms.

Regardless of which option you choose, it's important to foster a positive working relationship with your team. Provide regular feedback and support, and encourage open communication and collaboration to ensure the success of your project.

Ultimately, the key to successfully working with freelancers or in-house teams is to approach them as valuable members of your team and to leverage their skills and expertise to achieve your business goals.

### **References**

- <https://waydev.co/software-development-team-management/>
- <https://www.upwork.com/resources/manage-team-of-freelancers>

# Chapter 5: Tech Maintenance and Support

## 5.1 What is Software Maintenance & Software Support



## 5.1 What is Software Maintenance & Software Support

Software maintenance and software support are two related activities that are critical to ensuring the continued functioning and improvement of software applications.

Software maintenance entails continuously updating, fixing, and enhancing software after it has been released to users. It's like taking care of a pet – you need to ensure its health and wellbeing to keep it happy and functional. This may include bug fixes, security updates, performance improvements, and feature enhancements. The goal of software maintenance is to ensure that the software remains reliable, efficient, and effective over time and that it continues to meet the needs of its users.

Software support, on the other hand, refers to the activities that are undertaken to help users of software applications to troubleshoot issues, answer questions, and generally ensure that they are able to use the software effectively. This may include providing documentation, training, and technical support to users, as well as responding to requests for help and resolving issues that are encountered.

Both software maintenance and software support are important aspects of software development and deployment and require ongoing attention and resources in order to ensure that software applications continue to meet the needs of their users and remain useful over time.

## **5.2 Software Maintenance: What it Includes, Benefits, Cost, and Process**

Software maintenance is the process of modifying and updating the software after its initial release in order to ensure that it continues to meet the needs of its users and remains functional and effective over time. This can include activities such as bug fixes, security updates, performance improvements, and feature enhancements.

Some of the benefits of software maintenance include:

- **Improved reliability:** By addressing bugs and other issues, software maintenance can improve the reliability and stability of software applications.
- **Enhanced security:** Software maintenance can help to identify and address security vulnerabilities, reducing the risk of data breaches and other security incidents.



- **Increased performance:** By optimizing code and addressing performance issues, software maintenance can help to improve the speed and efficiency of software applications.
- **Improved user experience:** By adding new features and addressing usability issues, software maintenance can help to enhance the overall user experience of software applications.

However, software maintenance can also be a costly and time-consuming process, particularly if the software is poorly designed or lacks proper documentation. The cost of software maintenance can vary depending on factors such as the size and complexity of the software application, the nature of the updates required, and the resources available to the maintenance team.

The process of software maintenance typically involves several steps, including:

- **Problem identification:** Identifying issues and bugs that need to be addressed, often through feedback from users or testing processes.
- **Problem analysis:** Analyzing the issue to determine its root cause and the best way to fix it.
- **Solution design:** Designing a solution to address the issue, which may involve modifying code or adding new features.
- **Implementation:** Implementing the solution and testing it to ensure that it resolves the issue without causing any new problems.
- **Verification:** Verifying that the solution works as intended and that the software application is functioning correctly.

Overall, software maintenance is an essential process for ensuring the ongoing reliability and effectiveness of software applications and can help to ensure that software remains useful and relevant over time.

## 5.3 Software Support: What it Includes, Benefits, Cost, and Process

Software support is the process of providing assistance and resolving issues for users of software applications. This can include activities such as troubleshooting, answering questions, providing documentation, training, and technical support.

Some of the benefits of software support include:

- **Improved user satisfaction:** By providing timely and effective support, software support can help to improve user satisfaction and loyalty.
- **Increased productivity:** By resolving issues quickly and efficiently, software support can help to minimize downtime and improve productivity.
- **Better software performance:** By providing guidance on how to use software applications effectively, software support can help to improve overall performance and efficiency.
- **Enhanced user skills:** By providing training and resources, software support can help users to develop new skills and become more proficient in using software applications.

However, providing software support can also be a costly and time-consuming process, particularly if support needs are frequent or complex. The cost of software support can vary depending on factors such as the size and complexity of the software application, the nature of the issues encountered, and the resources available to the support team.

The process of software support typically involves several steps, including:

- **Issue identification:** Identifying issues reported by users, often through helpdesk tickets, email, or phone support.
- **Issue triage:** Assessing the severity and impact of each reported issue, and prioritizing them accordingly.
- **Issue resolution:** Resolving issues through various means, such as providing guidance, troubleshooting, modifying software configurations, or patching code.

- **Verification:** Verifying that the issue has been successfully resolved, and confirming that the software application is functioning correctly.
- **Documentation:** Documenting the issue, its resolution, and any relevant information for future reference.

Overall, software support is an essential process for ensuring that users are able to effectively use software applications, and can help to minimize frustration and lost productivity due to technical issues.

## 5.4 Software Maintenance and Support Setup Plan

A software maintenance and support setup plan is a comprehensive document that outlines the processes, resources, and procedures required to effectively maintain and support software applications over time. It is important to have a well-defined setup plan in place to ensure that software is reliable, efficient, and effective and that users are able to receive timely and effective support when needed.

Here are some key components that should be included in a software maintenance and support setup plan:

- **Maintenance and support goals:** Clearly outline the goals and objectives of the maintenance and support plan, such as improving reliability, addressing security vulnerabilities, and enhancing user experience.
- **Team structure:** Define the roles and responsibilities of the maintenance and support team members, including developers, testers, support staff, and project managers.
- **Maintenance processes:** Detail the processes for identifying, analyzing, and resolving software issues, including bug tracking, issue prioritization, solution design, implementation, and verification.
- **Support processes:** Outline the processes for providing technical support to users, including issue identification, triage, resolution, and documentation.

- **Service level agreements (SLAs):** Define the SLAs for maintenance and support activities, including response times, issue resolution times, and availability.
- **Tools and resources:** Identify the tools and resources required to support maintenance and support activities, such as bug-tracking software, documentation, and training materials.
- **Testing and quality assurance:** Describe the testing and quality assurance processes required to ensure that updates and enhancements are properly tested before being released to users.
- **Communication and reporting:** Establish a communication and reporting plan to ensure that stakeholders are kept informed of maintenance and support activities, including progress updates, issue resolutions, and service level adherence.
- **Budget and resource allocation:** Define the budget and resource requirements for maintenance and support activities, including personnel, equipment, and software.
- **Continuous improvement:** Establish a process for continuous improvement, including regular reviews of maintenance and support activities, feedback from users, and analysis of software usage data.

Overall, a well-defined software maintenance and support setup plan is critical for ensuring the ongoing reliability and effectiveness of software applications, and for providing users with the support they need to effectively use software. By following a well-defined plan, organizations can minimize downtime, improve user satisfaction, and ensure that software applications remain relevant and useful over time.

## 5.5 In-house software support & maintenance vs Outsourced software support & maintenance

When it comes to software support and maintenance, organizations have the option of handling these activities in-house or outsourcing them to a third-party provider. Here are some of the key differences between in-house and outsourced software support and maintenance:

### **In-House Software Support and Maintenance:**

- **Control:** With in-house support and maintenance, organizations have greater control over the processes, resources, and procedures used to maintain and support their software applications.
- **Familiarity:** In-house support and maintenance teams are likely more familiar with the organization's software applications and business processes, which can lead to faster issue resolution and better overall support.
- **Cost:** While in-house support and maintenance may provide greater control and familiarity, it can also be more expensive, as it requires a dedicated team of staff and resources.
- **Flexibility:** In-house support and maintenance can offer greater flexibility in terms of responding to unique or urgent support needs, as the team is able to quickly adapt and prioritize issues as needed.

### **Outsourced Software Support and Maintenance:**

- **Expertise:** Outsourced support and maintenance providers are often specialized in providing support services, and can bring a high level of expertise and knowledge to the table.
- **Cost:** Outsourced support and maintenance can be more cost-effective than in-house support and maintenance, as it allows organizations to tap into a pool of resources and expertise without having to maintain a dedicated team.
- **Scalability:** Outsourced support and maintenance can be more scalable than in-house support and maintenance, as providers have the ability to quickly ramp up or down resources as needed.
- **Focus:** Outsourced support and maintenance providers are solely focused on providing support and maintenance services, which means they can dedicate all of their resources and attention to these activities.

Ultimately, the decision to handle software support and maintenance in-house or to outsource these activities will depend on a variety of factors, including the organization's budget, the level of control desired, and the level of expertise

available in-house. Both options have their advantages and disadvantages, and organizations should carefully consider their specific needs and requirements before making a decision.

## **5.6 Are software maintenance and support essential?**

Yes, software maintenance and support are essential for ensuring the ongoing reliability, performance, and security of software applications. Software maintenance refers to the ongoing activities required to keep software applications up-to-date, address bugs and defects, and make necessary changes or updates to ensure compatibility with changing technology and user needs. Without proper maintenance, software applications can become outdated, and unreliable, and may even pose security risks.

Software support, on the other hand, is critical for providing users with the assistance they need to effectively use and troubleshoot software applications. Technical support staff can help users navigate issues, answer questions, and provide training and resources to help users get the most out of their software. Without effective support, users may struggle to use software applications effectively, leading to frustration and decreased productivity.

Overall, software maintenance and support are essential for ensuring the ongoing success of software applications. By investing in these activities, organizations can ensure that their software applications remain reliable, efficient, and effective over time, and can provide users with the support they need to effectively use and benefit from these applications.

### **References**

- <https://www.clariontech.com/blog/how-software-maintenance-and-support-services-can-help-you-beat-the-curve>
- <https://www.scnsoft.com/services/software-development/software-support-and-maintenance>
- <https://www.softude.com/blog/what-is-the-difference-between-software-maintenance-and-software-support>

## Conclusion

"An Entrepreneur's Guide to Develop Software for Business" is an essential resource for entrepreneurs who want to develop software for their businesses. It covers every aspect of the development process, from initial planning to ongoing maintenance and support. By following the steps outlined in this book, entrepreneurs can ensure that their software development process is smooth, efficient, and effective.

## Contact with QSL

### Qttec Solution Limited

---



+8801313 522 828



info@qttecsolution.com



House: 23, Road: 04, Nikunja-2,  
Khilkhet, Dhaka, Bangladesh



www.qttecsolution.com

