# **Feasibility Study**

**Introduction**

A feasibility study is a detailed analysis conducted to determine the viability of a proposed project or idea. It helps decide whether a project is feasible and worth pursuing by evaluating various factors like technical requirements, financial implications, and market potential. Here's a breakdown of the key components typically included in a feasibility study, especially for a software project

1. **Technical Feasibility**
   1. **Introduction**

The Technical Feasibility section examines the technological requirements and resources necessary for setting up and operating an online store. This includes the choice of platform, payment solutions, and technical infrastructure.

* 1. **Technical Requirements**

**1.2.1)** **Software Requirements**

1. Payment Gateway
2. Operating System like windows, mac or Linux
3. Database like MySQL
4. Web server
5. Development Environment like VS code and Git Hub
6. E-commerce Platform (optional)

**1.2.2)** **Hardware Requirements**

1. Servers: We can use cloud hosting platforms like AWS, Google Cloud, or Azure for servers, backups, and storage.
2. Internet Connectivity: High-speed internet connection essential for managing store operations and development
3. Development Devices: we need PCs or laptops to develop.
   1. **Technical Expertise**
4. Frontend: we use languages like (JavaScript, HTML5, CSS3) and use libraries like Bootstrap or Material-UI
5. Backend: For the backend, we will use native PHP
6. Database: We will use a relational database like MySQL.
7. UI/UX: use Canva to design pages
   1. **Project Complexity and Risks**
8. Scalability: Ensure the system can handle growth in users and data.
9. Security Requirements: Identify necessary cybersecurity measures to protect user data
10. Website Performance:

* Slow Loading: Pages may take too long to load, causing frustration.
* Downtime: The website could be temporarily unavailable, leading to lost sales.

1. Navigation Issues: The user interface must be simple to make it easy for users to navigate the website.
2. Account Management: Users may face challenges while managing an account or in setup.
3. Product Information: Lack of detailed product information can make users hesitant to purchase
4. **Economic Feasibility**

**2.1) Introduction**

Economic Feasibility helps determine if the financial benefits of a project justify the investment required. This is achieved through a cost-benefit analysis, ROI calculation, NPV calculation, break-even analysis, and risk assessment

**2.1) Identify and Estimate Costs**

* + **Initial costs**

|  |  |  |  |
| --- | --- | --- | --- |
| Cost Category | Item | Description | Initial Cost |
| Website Development & Design | Domain Name | Website address registration | 500 |
|  | SSL Certificate | Security certificate for HTTPS | 500 |
|  | Front-End Development | User interface (HTML, CSS, JavaScript) | 1,500 |
|  | Back-End Development | Server-side logic, databases | 2,000 |
|  | E-commerce Integration | Shopping cart, checkout, payment APIs | 500 |
|  | UI/UX Design | Users experience design & testing | 1,000 |
| E-commerce & Payment Solutions | Payment Gateway Fees | Fees for processing payments | 1.000 |
| Marketing & Advertising | Initial Marketing Campaign | First ads on social media, Google Ads | 1,000 |

|  |  |
| --- | --- |
| **Total** | **8000** |

* **Operational Costs**

|  |  |  |  |
| --- | --- | --- | --- |
| Cost Category | Item | Description | Operational Cost |
| Website Development & Design | Hosting | Website server hosting | 250 |
|  | Website Maintenance | Updates and fixes | 350 |
|  | E-commerce & Payment Solutions | Payment Gateway Fees | 100 (2.9% + 0.30 per transaction) |
| Marketing & Advertising | Initial Marketing Campaign | ads on social media, Google Ads | 500 |
|  | Social Media Management | Content creation and engagement | 450 |
| Customer Support | Part-time support | Support for inquiries | 350 |

|  |  |
| --- | --- |
| **Total** | **2000** (excluding transaction fees) |

**2.3)** **Identify and Estimate Benefits**

### **Estimated Revenue Calculation**

* Average Order Value:200$
* Monthly Orders:100
* Monthly Revenue: Average Order Value × Monthly Orders >>100\*200=20000$
* **Cost of Goods Sold**

50% of revenue, or $20,000 × 50% = $10,000

* **Gross Profit**

Gross Profit = Monthly Revenue – COGS

20,000-10,000=10,000

* **Net Profit**

Net Profit = Gross Profit - Operating Expenses

10,000-2000=8000

* **Annual Profit**

monthly net profit \* 12

8000\*12=96,000

|  |  |  |
| --- | --- | --- |
| Metric | Calculation | Amount |
| Monthly Revenue | Average Order Value × Monthly Orders | $20,000 |
| Cost of Goods Sold (COGS) | Revenue × COGS % (e.g., 50%) | $10,000 |
| Gross Profit | Monthly Revenue - COGS | $10,000 |
| Operating Expenses | Monthly costs (from cost table) | $2,000 |
| Net Profit | Gross Profit - Operating Expenses | $8,000 |
| Annual Profit | monthly net profit \* 12: | $96,000 |

## **2.4) Calculate Net Present Value (NPV)**

NPV helps determine the investment’s feasibility by calculating the present value of future cash flows and subtracting initial costs.

## Required Inputs

1. Future Cash Flows: Estimate the annual cash flow expected from the project.  
 - Assume the annual net profit of the project, after accounting for revenue and expenses, is $96,000.

2. Discount Rate: Used to account for inflation and investment risk over time.  
 - Let’s assume a discount rate of 10%.

3. Initial Costs: The total cost paid to start the project, representing the initial investment.  
 - Assume the initial cost of $8,000.

4. Project Duration: The expected number of years the project will continue or yield cash flows.  
 - Let’s assume the project will last for 5 years.

## NPV Calculation Formula

A black text on a white background

Description automatically generated

**A math problem with numbers and a few lines

Description automatically generated with medium confidence**- Cash Flow: Annual cash flow for year t  
- r: Discount rate  
- t: Year (from 1 up to project duration)

## **2.5)** **Calculate Return on Investment (ROI)**

A close up of black text

Description automatically generated

Inputs for ROI Calculation

1. Total Net Profit: This is the total cash flow over the project duration minus the initial investment.
   * In our case, the total cash flows over 5 years (from the NPV calculation) is $363,915.16.
   * Subtracting the initial investment of $8,000, the total net profit is:
   * 363,915.16−8,000=355,915.16
2. Initial Investment: The cost required to start the project, which is $8,000.

A number and a line

Description automatically generated with medium confidence

**2.6)** **Risk Assessment**

 **Market Size Misestimation**: Underestimating or overestimating the potential customer base can lead to inaccurate sales forecasts.

 **Revenue Projections**: Overly optimistic sales expectations may result in financial difficulties if actual sales fall short.

 **Unexpected Costs**: Unforeseen expenses, such as website maintenance or increased shipping costs, can negatively impact profit margins.

 **Customer Acquisition Costs**: High costs associated with attracting customers through marketing and advertising can reduce overall profitability.

 **Inventory Management Risks**: Poor inventory control can lead to excess stock, resulting in increased holding costs, or stockouts, leading to missed sales opportunities.

 **Payment Processing Fees**: Transaction fees from payment processors can significantly reduce profit margins, especially with high sales volumes.

 **Economic Downturns**: Economic challenges can decrease consumer spending, directly impacting sales and revenue.

 **Supply Chain Disruptions**: Issues with suppliers or logistics can lead to delays in product availability, affecting sales and customer satisfaction.

 **Legal Compliance Risks**: Failing to comply with e-commerce regulations, such as data protection laws, can result in fines and legal issues.

 **Competitive Pressure**: A saturated market with many competitors can make it difficult to attract and retain customers, impacting profitability.

## **Organizational Feasibility**

**3.1) Introduction**

Organizational feasibility analysis is essential for assessing whether your online store project is viable within your team's structure and resources.

**3.2) Project Objectives and Goals**

**Vision**

To create an easy-to-use online store where customers can browse and buy products comfortably and conveniently

**Mission**

To offer high-quality products, smooth shopping experience, and reliable customer support to make online shopping simple and enjoyable while ensuring their information is safe.

**Goals**

1. **User Growth**
   * **Goal**: Reach 1,000 active users each month within the first 6 months.
   * **Plan**:
     + Promote our store through social media, email, and partnerships to attract visitors.
     + Encourage users to refer to friends through rewards.
     + Adjust marketing based on user feedback to improve reach.
2. **Customer Satisfaction**
   * **Goal**: Achieve an 85% or higher satisfaction rating within the first year.
   * **Plan**:
     + Provide friendly and responsive customer support.
     + Make returns and refunds easy to build trust.
     + Regularly ask customers for feedback to keep improving the shopping experience.
3. **Sales Targets**
   * **Goal**: Generate $10,000 in monthly sales within the first year.
   * **Plan**:
     + Offer competitive prices and discounts to encourage purchases.
     + Run special promotions for holidays and returning customers.
     + Use product recommendations to boost the average order value.
4. **Efficient Operations**
   * **Goal**: Improve order accuracy and reduce processing time by 20%.
   * **Plan**:
     + Streamline our order and inventory process to keep everything updated and organized.
     + Train the team on effective order management to reduce errors.
5. **Brand Building**
   * **Goal**: Build a well-known, trusted brand in the first year.
   * **Plan**:
     + Develop a strong brand identity with a recognizable logo and style.
     + Share helpful content like blogs or guides to engage customers.
     + Grow our social media presence and connect with the community.
6. **Security & Reliability**
   * **Goal**: Ensure a secure platform with 99% uptime.
   * **Plan**:
     + Protect user data with secure payment methods.
     + Keep the site updated to ensure it’s safe and always available.
     + Make the store mobile-friendly for easy shopping on any device.

**3.2) Skills and Roles Needed**

1. Project Leader (Mohamed Ali)

* Responsibilities:
  + Oversee the entire project from start to finish.
  + Assign tasks to team members and make sure everyone is on track to meet deadlines.
  + Lead weekly meetings to discuss progress, address challenges, and plan the next steps.

2. Back-end Development (Mohamed Ali and Mohamed Abd El Salam)

* Responsibilities:
  + Build and maintain the server-side of the application where data is processed.
  + Manage databases to store and retrieve product information and user data.
  + Create APIs (Application Programming Interfaces) that allow the front-end to communicate with the back end.
* Skills Needed: Knowledge of programming languages like Node.js or Python, and experience with databases like MongoDB or PostgreSQL.

3. Front-end Development (Mohamed El Nagar, Fathy and Mohamed Atef)

* Responsibilities:
  + Create the user interface (UI) that customers interact with on the website.
  + Ensure the website looks good and works well on both mobile and desktop devices.
  + Use frameworks like React or Vue.js to build interactive features.
* Skills Needed: Experience with HTML, CSS, JavaScript, and familiarity with front-end frameworks.

4. UI/UX Design (Mohamed Henish and Mahmoud Ali)

* Responsibilities:
  + Design a simple and attractive layout for the website.
  + Create wireframes (basic sketches of the website layout) and prototypes to visualize the design.
  + Conduct user testing to gather feedback on the design and improve usability.
* Skills Needed: Proficiency in design tools like Figma or Adobe XD, and an understanding of user-centered design principles.

5. Quality Assurance (Mohamed Abd El Salam, Mahmoud Ali and Mohamed Henish)

* Responsibilities:
  + Test the website as features are developed to find and fix bugs.
  + Ensure that the website works as intended and meets user needs before launch.
  + Use testing tools like Jest for front-end testing and Postman for checking APIs.
* Skills Needed: Attention to detail, understanding of software testing processes, and experience with testing tools.

6. Customer Support Preparation

* Responsibilities:
  + Create helpful resources like FAQs (Frequently Asked Questions) and user manuals to assist customers.
  + Set up a feedback system to gather customer opinions after the site goes live.
  + Initially, this role can rotate among team members, or you may consider hiring someone to handle customer support full-time later.
* Skills Needed: Good communication skills and the ability to empathize with customers' needs.

Summary of Team Structure

* Team Leader: Mohamed Ali
* Back-end Developers: Mohamed Ali and Mohamed Abd El Salam
* Front-end Developers: Mohamed El Nagar, Fathy and Mohamed Atef
* UI/UX Designer: (Mohamed Henish and Mahmoud Ali
* Quality Assurance Tester: Mohamed Abd El Salam and Mohamed Ali
* Customer Support: Shared responsibility among team members, potentially outsourced later.

**3.3) Team Availability and Skill Assessment**

#### **1. Assessing Availability**

To ensure the project runs smoothly, it’s important to know how much time each team member can dedicate to the project each week. Here’s how to do it:

* **Weekly Planning Meetings**
  + **Duration**: Schedule a 1-hour meeting each week.
  + **Purpose**:
    - Set goals for the week.
    - Review what was accomplished in the previous week.
    - Identify and address any obstacles or challenges team members are facing.
  + **Frequency**: Weekly, on a set day and time that works for everyone.
* **Weekly Development Hours**
  + **Individual Contributions**: Each team member should commit to a certain number of hours per week based on their schedules and other commitments.
  + **Suggested Hours**: Aim for about 5-10 hours each week for development work. This can vary based on individual availability.
  + **Adjustment**: If someone can contribute more or less time, adjust timelines and responsibilities accordingly.

#### **2. Example of Availability Assessment**

To assess availability, each team member should communicate their preferred meeting times and the number of hours they can contribute weekly:

* **Mohamed Ali (Leader)**: Available for 5-8 hours a week, mostly in the evenings.
* **Mohamed El Nagar**: Can commit 10 hours, has a flexible schedule.
* **Mohamed Atef**: Can contribute 5 hours, mostly available on weekends.
* **Mohamed Henish**: Available for 5-8 hours, usually in the evenings.
* **Fathy**: Can contribute 5 hours, available weekdays.
* **Mahmoud Ali (Designer)**: Has 5-8 hours, works part-time in the mornings.
* **Mohamed Abd El Salam**: Available for 10 hours, has flexible hours.
* **Team Member 8**: Can contribute 5 hours, available in the evenings.

### **3. Workflow Management**

* **Daily Check-ins**: Encourage brief daily check-ins via messaging apps (like Slack or Teams) for quick updates and support.
* **Task Management Tools**: Use tools like Trello, Asana, or Jira to assign tasks, track progress, and keep everyone informed about deadlines.

### **4. Timeline Adjustments**

* Based on the availability and contributions from each member, adjust project timelines to ensure that goals are realistic and achievable.
* If certain team members can’t contribute as much, consider redistributing tasks to balance the workload.

**3.4) Management Structure and Workflow**

 **Weekly Meetings**: Schedule a weekly project status meeting led by Mohamed Ali to review progress, assign tasks, and address blockers.

 **Task Management**: Use GitHub Projects or an alternative like Trello or Asana for task tracking.

 **Reporting**: Team members report their progress weekly, with any critical issues flagged for immediate attention.

**3.5) Technology and Resource Requirements**

#### **1. Development Tools**

* **Back-end Development**:
  + **Languages**: Use PHP for server-side logic.
  + **Database Management**: Use MySQL for managing relational databases.
* **Front-end Development**:
  + **Languages**: Utilize HTML, CSS, and JavaScript to build the user interface.

#### **2. Design Tools**

* **UI/UX Design**:
  + Use Figma for creating wireframes, prototypes, and design elements.

#### **3. Testing Tools**

* **Quality Assurance**:
  + Utilize browser developer tools for testing the user interface and functionality.

#### **4. Hosting and Deployment**

* **Web Hosting**:
  + Choose a reliable hosting provider like AWS or Digital Ocean, or use a local server.
* **Domain Name**:
  + Register a domain name that reflects your brand for easier customer access.

#### **5. Collaboration and Communication Tools**

* **Project Management**:
  + Use tools like Trello, Asana, or Jira for assigning tasks and tracking progress.
* **Communication**:
  + Set up channels on Slack or Microsoft Teams for daily updates among team members.

#### **6. Customer Support Resources**

* **Support Documentation**:
  + Prepare FAQs and user manuals to assist customers’ post-launch.
* **Feedback Tools**:
  + Implement tools like Google Forms or Type form for collecting customer feedback.

#### **7. Security Measures**

* **SSL Certificate**:
  + Obtain an SSL certificate to ensure secure connections for online transactions.
* **Data Protection**:
  + Implement strong authentication methods and data encryption to protect user information.

#### **8. Hardware Requirements**

* **Development Machines**:
  + Ensure each team member has a computer with adequate specifications:
    - **Processor**: Intel i5 or AMD Ryzen 5 (or better).
    - **RAM**: At least 8 GB (16 GB recommended).
    - **Storage**: SSD with a minimum of 256 GB for faster performance.
* **Testing Devices:**
  + Access to various devices (smartphones, tablets, different browsers) is essential for compatibility testing.
* **Internet Connectivity**: High-speed internet connection essential for managing the store

**3.6) Operational Plan**

1. **Sprints cycles** 
   * **Duration**: Two weeks.
   * **Goal**: Build key features quickly.
2. **Key Features**:
   * **User Registration**: Users create accounts.
   * **Product Catalog**: List of products.
   * **Shopping Cart**: Add and remove items.
   * **Checkout**: Review and pay for items.
   * **Payment**: Use secure payment methods (like PayPal).

**Code Review**

* **Peer Review**: Team members check each other’s code on GitHub for quality.
* **Frequency**: Do reviews regularly during sprints.

**Testing**

* **Manual Testing**: Check features by hand.
* **Automated Testing**: Use tools to test parts of the code automatically.
* **Types of Tests**:
  + **Unit Testing**: Test single parts.
  + **Integration Testing**: Check how parts work together.
  + **User Testing**: Get feedback from users before launch.

**Customer Support**

1. **Support Options**:
   * **Email**: Create a support email
   * **Live Chat**: Add a chat feature on the website.
2. **Handling Questions**:
   * Decide who will answer customer inquiries.
   * Prepare quick responses for common questions.

**3.7) Risk Assessment and Mitigation Strategies**

#### **1. Identify Risks**

* **Technical Risks**:
  + **Issue**: Bugs or performance problems in the code.
  + **Mitigation**: Implement regular code reviews and testing throughout development.
* **Security Risks**:
  + **Issue**: Data breaches or cyberattacks.
  + **Mitigation**: Use secure coding practices, SSL certificates, and data encryption.
* **Resource Risks**:
  + **Issue**: Team members may be unavailable due to personal reasons or workload.
  + **Mitigation**: Cross-train team members and create a backup plan for key roles.
* **Market Risks**:
  + **Issue**: Changes in customer preferences or competition.
  + **Mitigation**: Conduct regular market research and gather customer feedback to stay updated.
* **Project Management Risks**:
  + **Issue**: Delays in the project timeline.
  + **Mitigation**: Set clear deadlines, hold regular check-ins, and adjust tasks as needed.

#### **2. Monitor Risks**

* **Regular Reviews**: Schedule periodic meetings to discuss and evaluate potential risks.
* **Feedback Loop**: Encourage team members to report any new risks or concerns they identify during development.

#### **3. Respond to Risks**

* **Action Plans**: For each identified risk, develop a clear action plan outlining how to respond if the risk occurs.
* **Flexibility**: Be prepared to adapt your strategy based on changing circumstances or new information

**3.8)**

#### **1. Future Growth Ideas**

* **New Features**: Plan to add features based on what customers want, like:
  + Better search options.
  + Personalized product recommendations.
  + Loyalty programs.
* **Expand Market**: Investigate reaching new customers by:
  + Offering your site in different languages.
  + Tailoring marketing to specific regions.
* **Partnerships**: Team up with other businesses to attract new customers.

#### **2. Continuous Improvement**

* **Customer Feedback**: Regularly ask for feedback to understand how to make the shopping experience better.
* **Performance Tracking**: Use tools to check how well your site is doing and see where improvements can be made.

## **4. Schedule Feasibility**

## 1. Introduction

Schedule Feasibility assesses the proposed timeline to determine if the project can be completed within the desired timeframe.

## 2. Project Timeline and Phases

The project is divided into five primary phases with optimized durations to meet the 10-week schedule:

* **Phase 1: Requirements Gathering and Analysis** (1 week)
  + Conduct a rapid needs assessment with stakeholders.
  + Define project scope, finalize features, and establish technical specifications.
  + Deliverables: Comprehensive requirements document, initial project roadmap.
* **Phase 2: Design and Development** (4 weeks)
  + **Weeks 2-3:** Design user interface and user experience (UI/UX) for the store.
    - Focus on wireframes, prototyping, and user feedback to ensure usability.
  + **Weeks 3-5:** Front-end and back-end development, including database setup, product catalog, and shopping cart functionality.
  + **Week 5:** Integration of essential e-commerce features such as product search, reviews, and user accounts.
  + Deliverables: Approved UI/UX design, functional development build.
* **Phase 3: Integration and Testing** (2 weeks)
  + **Weeks 6-7:** Integrate payment gateways and third-party services (shipping, tax calculation).
  + Conduct comprehensive testing, including functionality, security, and usability.
  + Deliverables: Fully integrated e-commerce system, complete with initial testing results.
* **Phase 4: Marketing and Launch Preparation** (2 weeks)
  + **Weeks 8-9:** Finalize digital marketing strategy, including SEO, social media, and content.
  + Prepare launch campaign materials, press releases, and outreach strategy.
  + Deliverables: Marketing collateral, campaign calendar, customer support training.
* **Phase 5: Launch and Post-Launch Support** (1 week)
  + **Week 10:** Launch the website, actively monitor for issues, and ensure prompt responses to customer inquiries.
  + Provide initial customer support and monitor performance metrics.
  + Deliverables: Successful public launch, performance tracking setup, active support channels.

## 3. Milestones and Deliverables

To maintain the 10-week schedule, the following milestones are crucial:

* **Milestone 1**: Requirements Document Completed (End of Week 1)
* **Milestone 2**: UI/UX Design Approved (End of Week 3)
* **Milestone 3**: Development Phase Completion (End of Week 5)
* **Milestone 4**: Completed Integration and Testing (End of Week 7)
* **Milestone 5**: Marketing Campaign Ready (End of Week 9)
* **Milestone 6**: Successful Website Launch (End of Week 10)

## 4. Risk Assessment for Schedule Feasibility

To meet the accelerated schedule, potential scheduling risks and mitigation strategies include:

* **Development Delays**
  + *Risk*: Unanticipated technical challenges could delay the development phase.
  + *Mitigation*: Implement agile development practices with daily stand-ups and weekly sprints to quickly address issues.
* **Testing and Debugging Overruns**
  + *Risk*: Testing may take longer, especially for critical e-commerce functionalities like payment processing.
  + *Mitigation*: Allocate extra time for testing in each sprint and prioritize automated testing for high-traffic features.
* **Marketing and Content Delays**
  + *Risk*: Delay in finalizing marketing content could impact pre-launch activities.
  + *Mitigation*: Begin marketing preparation alongside development, ensuring content is ready in parallel with the main building.
* **Resource Availability**
  + *Risk*: Limited availability of team members or dependencies on third-party integrations.
  + *Mitigation*: Crosstrain team members for flexibility and coordinate closely with third-party providers to avoid bottlenecks.

## 5. Conclusion on Schedule Feasibility

This analysis indicates that, with clear milestones, disciplined time management, and proactive risk mitigation, the Online Store Project is feasible within a 10-week timeframe. The project’s success relies on effective communication, agile practices, and early identification of issues, allowing the team to meet the launch deadline with a fully functional and optimized online store.