Analysis

Introduction to analysis

Analysis is the process of defining the needs and necessity of certain functions of the users for an application that is to be built. Analysis is one of the important stages of the software development process. Analysis helps to determine the feasibility of the project. Analysis is the process of collecting data, finding requirements for the project, problems in the system to insure a feasible new system.

Conducting the analysis to produce the best results for the decisions to be made is an important part of the process, as is appropriately presenting the results. It is very important to not overcomplicate the analysis and to clearly and succinctly present results.

There are various types of analysis that can be conducted such as PEST analysis, SWOT analysis, CATWOE analysis and more. For this project I applied SWOT analysis

SWOT analysis

What makes SWOT particularly powerful is that, with a little thought, it can help you uncover opportunities that you are well-placed to exploit. And by understanding the weaknesses, you can eliminate threats that would otherwise harm the company. You can start to craft a strategy that helps you distinguish yourself from your competitors, so that you can compete successfully in your market.



*Fig 1: SWOT*

Strengths

What advantages does your organization have?

What do you do better than anyone else?

Weaknesses

What could you improve?

What should you avoid?

Opportunities

What good opportunities can you spot?

What interesting trends are you aware of?

Threats

What obstacles do you face?

Could any of your weaknesses seriously threaten your business?

Feasibility study

A feasibility study is an analysis that takes all of a project's relevant factors into account—including economic, technical, legal, and scheduling considerations to complete the project successfully. Feasibility studies also can provide a company's management with crucial information that could prevent the company from falling apart.

Types of feasibility study that are performed during analysis are:

Economic feasibility

Generally, it means whether a business or a project feasible cost wise. A project is considered economically feasible when profit can be observer for the project.

Technical feasibility

In technical feasibility issues such as; - whether the required technology is available or not, whether the required resources like software, hardware, programmers and testers are available or not.

Schedule feasibility

**Schedule Feasibility** is defined as the probability of a project to be completed within its scheduled time limits, by a planned due date. If a project has a high probability to be completed on-time, then its schedule feasibility is appraised as high.

Operational Feasibility

Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented. Whether there will be resistance from users that will affect the possible application benefits.

Legal Feasibility

It includes study concerning contracts, liability, violations, and legal other traps frequently unknown to the technical staff.

Requirement analysis

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. Requirements analysis involves all the tasks that are conducted to identify the needs of different stakeholders as different stakeholders might have different ideas and needs.

Functional:

Functional requirements describe the desired end function of a system operating within normal parameters, so as to assure the design is adequate to make the desired product and the end product reaches its potential of the design in order to meet user expectations. The system will have a proper interface, authentication.

Non-functional:

Nonfunctional requirements are vital to the success of software systems. **Non-functional requirements describe how the system works.** There are **four examples of Non-Functional requirement groups**; reusability, performance, usability, and security.

Reusability:

Any system should be designed in such a way that it’s code could be re-used in any further programs that may come in future.

Usability:

Prioritize the important functions of the system based on usage patterns.  
**Frequently used functions should be tested for usability.**

Reliability:

**Users have to trust the system**, even after using it for a long time. It’s a good idea to also include requirements that make it easier to monitor system performance.

Security:

These are security related architectural requirements, like robustness or minimal performance and scalability. This requirement type is typically derived from architectural principals and good practice standards.

Moscow Priotization

MoSCoW prioritization, also known as the MoSCoW method or MoSCoW analysis, is a popular prioritization technique for managing requirements. The method is commonly used to help key stakeholders understand the significance of initiatives in a specific release. MoSCoW, stands for 4 different categories: must-haves, should-haves, could-haves, and won’t have this time.

Must have: Function which are necessary such as price list are at the top of priority list and will be included in the first version.

Should have: Function such as login that determine the person that sold the goods are listed as should have.

Could have: function such as “change language” could be added in future updates.

Won’t have: functions such as” play music” won’t be added any time soon.

|  |  |  |
| --- | --- | --- |
| Id | Title | Priority |
| F001 | Price list | Must have |
| F002 | Login | Should have |
| F003 | Change language | Could have |
| F004 | Music player | Won’t have |

*Table: Moscow Priotization*

Use case:

Use case is a methodology used on system analysis to identify and clarify the requirements