Quest Journals
Journal of Software Engineering and Simulation
Volume 8 ~ Issue 10 (2022) pp: 10-13
ISSN(Online) :2321-3795 ISSN (Print):2321-3809
www.questjournals.org

Research Paper

Functional and Non Function Requirements of the Traffic Flow Prediction System

Tahseen[1]Dr. G.N.R. Prasad[2]
[1] *MCA IV Semester, Department of master of computer applications*
Chaitanya BharathiInstitute of Technology (A), Gandipet, Hyderabad – 500 075.
[2]*Sr. Asst. professor, Department of master of computer applications*
Chaitanya BharathiInstitute of Technology (A), Gandipet, Hyderabad – 500 075.

Abstract: *In the post covid-19 the business is started to move in a smooth way. They are some ups and downs because of the war in between Russia and Ukraine. The impact of war is very less in a strong economy countries like India, China and other some countries. When more commuters are using the various ways of transport system to move is the meaning of the economy. In India the traffic congestion is increasing every day. One of the major issues with traffic management and guiding systems as well as one of the crucial duties of an intelligent transportation system is traffic flow prediction. It is now feasible to assess and anticipate traffic conditions in smart cities more quickly and precisely thanks to the rapid proliferation of machine learning new methodologies and the advent of new data sources. Effective traffic congestion reduction and road capacity improvement are possible with the use of traffic estimating and forecast systems. The success of a system or software project may be evaluated thanks to the extremely important procedure known as requirements analysis. Most requirements fall into one of two categories:Requirements that are functional and non-functional. In this study, the issues and difficulties of the prediction models are thoroughly examined, as well as the current traffic forecast approaches for smart cities. The potential future development trend of short-term traffic flow predicting approaches is highlighted based on the examination of the current short-term traffic flow forecasting methodologies.*

Keywords: *Functional requirements , Machine learning, Non-functional requirements, Requirement Analysis, Software Engineering*

R*eceived 14 Oct., 2022;* ***R****evised 26 Oct., 2022;* ***A****ccepted 29 Oct., 2022 © The author(s) 2022.*
P*ublished with open access at* ***www.questjournals.org***

I. INTRODUCTION

Software that has been correctly built is necessary for computers, programmes, and websites to carry out the duties we want them to. An essential facet of the information technology sector that calls for training and expertise is software development. An explanation of the service that the programme must provide is contained in a functional requirement (FR). It describes a piece of software or a software system. A function is nothing more than the inputs, behaviour, and outputs of the software system. A system's likely purpose can be determined by a computation, data manipulation, business process, user interaction, or any other specialised feature. In software engineering, functional specifications are another name for functional requirements. A product ought to function effectively and have recognisable characteristics. You can create a unique product with non-functional criteria. You may create a system that satisfies the requirements of your end users by understanding instances of non-functional requirements and how they function in an application.

II. LITERATURE COLLECTION

Software development methodologiesare several approaches used in software development that are designed to assist the programme across the whole software development life cycle. The demands and tastes of the project and its creators are taken into consideration while selecting these models. The demands of the companies, the developer's preferred designs, the technique selected to deploy the software into production, and whether or not the method should be able to handle maintenance are some of the key factors when picking a method. The various software development methodologies exist and some of them are :

- The waterfall model
- Prototype model

- Agile software development
- Rapid application development
- Joint application development
- The fountain model
- The spiral model
- Open-source software development

The waterfall model is considered a predictive methodology and is the original SDLC method.
Any software development methodology should follow the following steps:

1. Identifying the requirements for the given software.
2. Analyzing the requirements using the various gathering techniques of the chosen software.
3. Detailing the specifications and requirements of the software.
4. Designing (Both physical and logical) the software.
5. Write the coding and programming.
6. Test the software.
7. Conducting maintenance activities on the finalized design.

Software development, often known as application development or software design, is a group of repetitive procedures used to plan, design, produce, distribute, and maintain software. This is accomplished through the use of several programming languages and coding, which provide the generated software its structure and functionality. Software may be made to fulfil a variety of personal and professional demands, goals, objectives, and procedures. Usually, this development is carried out in a number of organised stages.

2.1 Existing System:

The use of tiny sample data in prediction research is a result of the difficulty in obtaining early traffic data as well as the limited and poor quality of the data environment. As a result, a lot of prediction studies have adopted the complicated model of pure mathematics theory, which overlooks the fundamental traits and evolutionary process of traffic flow. However, too sophisticated models, including neural networks and combination models, involve intricate computations and complex processes, which make them unsuitable for use in actual short-term traffic forecast. The main limitation of the existing system ishuman and natural factors, such as communication problems, equipment failures are some of them.

III. DESIGN OF THE SYSTEM

2.2. Proposed system

In this paper, to prepare the final system various requirements analysis is studied.

3.2.1 The various Modules of the system are

- ❖ Insert new traffic data
- ❖ Trigger the training process
- ❖ Input data
- ❖ Get predicted traffic status

IV. REQUIREMENTS ANALYSIS

4.1 FUNCTINAL REQUIREMENTS (FR)

These are the specifications that the system must meet in order to satisfy the end user's fundamental needs. As a requirement of the contract, all of these features must be built into the system. These are shown or described as the input to be provided to the system, the operation carried out, and the intended outcome. In contrast to non-functional needs, they are essentially the user-stated criteria that are visible in the finished product.

4.1.1 Various Types of Functional Requirements

Here are the most common functional requirement types are used in development of the system.

- Business Rules
- Transaction Handling
- Certification Requirements
- Administrative functions
- Reporting Requirements
- Authorization levels
- External Interfaces
- Audit Tracking

- Legal and Regulatory Requirements
- Historical Data management

Here, some of the requirements belonging to the functional category are identified.

Functional Requirement Number	Description
FRN 1	User should be able to raise the question
FRN 2	Traffic report should be generated to the traffic police once in 15 minutes
FRN 3	An API interface to the system

Table 1 : Functional Requirements

Functional Requirements of a system is using following things:
• Information about each screen's actions.
• The system should have data handling logic.
• It ought to provide summaries of system reports or other outputs.
• Detailed information on the workflows that the system executes.
• It must expressly state who will be permitted to add, edit, and delete data in the system.
• The functional document has to provide information on how the system will satisfy any applicable regulatory and compliance requirements.

4.2 NON-FUNCTIONAL REQUIREMENT (NFR)

The quality attribute of a software system is specified by NFR. They assess the software system according to non-functional criteria such as responsiveness, usability, security, portability, and other criteria that are essential to the software system's success. "How quickly the programme loads" is an example of a non-functional need. Systems that don't meet non-functional criteria may not be able to meet user demands. You can place limits or limitations on the system architecture across different agile backlogs using non-functional requirements. For instance, when there are more than 1000 simultaneous users, the programme should load in 2 seconds.

Description of non-functional requirements is just as critical as a functionalrequirement.

- Usabilityrequirement
- Serviceabilityrequirement
- Manageabilityrequirement
- Recoverabilityrequirement
- Securityrequirement
- Data Integrity requirement
- Capacityrequirement
- Availabilityrequirement
- Scalabilityrequirement
- Interoperabilityrequirement
- Reliabilityrequirement
- Maintainabilityrequirement
- Regulatoryrequirement
- Environmentalrequirement
- Evaluate nonfunctional requirements

V. CONCLUSION

The identification of Functional Requirement allows the development team to prepare a quality software.The research made it possible to determine the benefits and drawbacks of producing a typical functional requirement document. assists you in determining if the application offers each of the features listed in the application's functional requirements. You can specify the functionality of a system or one of its subsystems using a functional requirement document. Identification of unmet requirements is aided by functional requirements and requirement analysis. They assist in precisely defining the desired system behaviour and functionality. The least expensive fixes are for errors discovered during the functional requirement collection stage. Encourage user objectives, tasks, or activities

REFERENCES

[1]. Fei-Yue Wang et al. Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications. IEEE Transactions on Intelligent Transportation Systems, 2010.
[2]. Yongchang Ma, Mashrur Chowdhury, MansourehJeihani, and Ryan Fries. Accelerated incident detection across transportation networks using vehicle kinetics and support vector machine in cooperation with infrastructure agents. IET intelligent transport systems, 4(4):328–337, 2010.
[3]. RutgerClaes, Tom Holvoet, and Danny Weyns. A decentralized approach for anticipatory vehicle routing using delegate multiagent systems. IEEE Transactions on Intelligent Transportation Systems, 12(2):364–373, 2011.
[4]. MehulMahrishi and SudhaMorwal. Index point detection and semantic indexing of videos - a comparative review. Advances in Intelligent Systems and Computing, Springer, 2020.
[5]. Joseph D Crabtree and NikiforosStamatiadis. Dedicated short-range communications technology for freeway incident detection: Performance assessment based on traffic simulation data. Transportation Research Record, 2000(1):59–69, 2007.
[6]. https://www.indeed.com/career-advice/career-development/non-functional-requirements-examples
[7]. https://www.modernrequirements.com/blogs/pillar/what-are-non-functional-requirements/
[8]. http://www.eiti.uottawa.ca/~bochmann/SEG3101/Notes/SEG3101-ch3-4%20-%20Non-Functional%20Requirements%20-%20Qualities.pdf