### 

### Fast IP Routing In Wireless Sensor Network

### Software Requirements Specification

### Version 1.0

### Prepared by

Pallavi Aithal N

Vaishnavi B

Contents

[Description](#_Toc77673482) 3

[Product Functions System Decomposition View](#_Toc77673484) 4

[User Characteristics](#_Toc77673485) 4

[Constraints](#_Toc77673486) 4

[Assumptions and Dependencies](#_Toc77673487) 4

[External Resource Requirements](#_Toc77673488) 4

[User Interface](#_Toc77673489) 4

[Hardware Interface](#_Toc77673490) 5

[Software Interface](#_Toc77673491) 5

[Communications Interface](#_Toc77673492) 5

[Functional Requirements Views](#_Toc77673493) 5

[Primary List of Functional Requirements](#_Toc77673494) 6

[<Req\_1>](#_Toc77673495) 6

[<Req\_2>](#_Toc77673496) 6

[<Req\_3>](#_Toc77673497) 6

[<Req\_4>](#_Toc77673498) 6

[<Req\_5>](#_Toc77673499) 6

[<Req\_6>](#_Toc77673500) 7

[<Req\_7>](#_Toc77673501) 7

[<Req\_8>](#_Toc77673502) 7

[Nonfunctional Requirements](#_Toc77673577) 8

[Availability](#_Toc77673580) 8

[Security](#_Toc77673581) 8

[Maintainability](#_Toc77673582) 8

# Description

The Fast IP Routing In WSN Software Requirements Specification provides a list and description of the functional and non-functional specifications for the software components of FIRWSN. It is derived in part from the RFC5714 Document and is supplemented by the FIRWSN Software Design Specification, the FIRWSN Software Test Plan.

* The FIRWSN Software Requirement provides the following views of the requirements.
* A list of the requirements presented numerically.
* A preliminary component grouping of the requirements.

A preliminary mapping of the requirements against proposed objects and their responsibilities.

The document is intended to establish the initial scope of the development effort.

## Product Functions System Decomposition View

Preliminary functional components of the product are as follows:

Stripe 1 : Start connection

* Stripe 2: Cmd-line Opening
* Stripe 2-a: Adding Route
* Stripe 2-b: Observe Failure
* Stripe 2-c: Modify Route
* Stripe 2-d: Repair Path
* Stripe 3: Fast Routing
* Stripe 4: Change Configuration
* Stripe 5: Save Changes
* Stripe 6: WSN Monitoring

Stripe 7: Options

The components mentioned above are more fully explored below, under the heading “Functional Requirements Component View.”

## User Characteristics

This software is designed for providing fast routing using failure detection methodology within less amount of time making users to access their required data even faster. However, its features offer pre-calculating the failure which leads to bypass the failed path or loop with minimal overhead.

The security methods which are currently present will be implemented on RFC5714 which cannot have intrusions and avoids hacking of user data or passwords.

## Constraints

The software serves largely to illustrate implemented activities on RFC5714. While, it includes a wide range of interesting features, such as command-line editing, editing configuration manually (Static routing) and automatic configuration (Dynamic routing).

It reduces the complexity of updating routing table of failed node or router every time when that server is down.

Since, helping users to understand the problem statement which we are solving easily by using backup path.

## Assumptions and Dependencies

The interface must be a single wireless network interface which can be used at a time.

If there are multiple wireless network interface available, then it depends on the user to select their required relevant interface.

# External Resource Requirements

The issues dealt with in this section of the document pertain to user and system requirements that affect the operation of the software within a given system or within a network context.

## User Interface

The user interface is based on components created using Command-line interface. The software runs on a personal computer. Command-prompt is used as user interface.

## Hardware Interface

The project executes on a PC equipped with Pentium CORE i3/i5/i7 that can run on Linux OS.

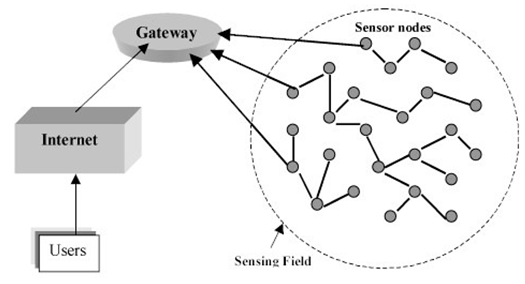
## Software Interface

Ns2 or ns3 simulators software on Linux.

## Communications Interface

The user interacts with the software using the mouse and the keyboard.

**ROUTING MECHANISM**



# Functional Requirements Views

This section provides the first of three views of the functional requirements for FIRWSN.

* The first view provides a primary list of the functional requirements. This list is organized sequentially.
* The second view provides a list of the functional requirements as grouped according to component. A component is also called a “stripe.” Each stripe represents a coherent grouping of functionality.

# Primary List of Functional Requirements

<Req\_1>

Software shall have the capability to add the route using ns2.

<Req\_2>

Software shall have the capability to modify the inserted route.

<Req\_3>

Software shall have the ability to provide user routing information in brief.

<Req\_4>

Software shall allow the user-interface to change the configuration.

<Req\_5>

Software shall have multi-hop failure avoidance mechanism.

<Req\_6>

Software shall have the ability to provide user routing information in brief.

<Req\_7>

Software shall provide security for the packets being transmitted.

<Req\_8>

Software shall have the ability to find micro loop and resolve the failure in routing.

# Functional Requirements Component View

This view of the requirements is dealt with in much greater detail in the FIRWSN Software Design Description. The Software Design description provides a componential view of the classes that implement the functionality. The breakdown here is meant only as a preliminary exploration of the scope of the software.

## Stripe 1—Start Connection

Requirement 1

## Stripe 1.1—Cmd-prompt

Requirement 1,2, 3,

## Stripe 2—Insertion

Requirement 1.

## Stripe 2.1—Modification

Requirements 4,5

## Stripe 3—Failure Observation

Requirement 4,6,7

## Stripe 3.1— Back-up Path

Requirement 6,7

## Stripe 3.2—Fast Routing

Requirements 5,6,7

## Stripe 4—Configuration

Requirements 4,5

## Stripe 5—Save Changes

Requirements 4,5,6,7

## Stripe 6— Monitoring

Requirements 6,7

NON-FUNCTIONAL REQUIREMENTS:

## Availability

The FIRWSN is available for the users to access the required data more faster compared to other existing algorithms.

## Security

There are no security issues as we are using other security methods that are already present in networking field.

## JITTER CONTROL

Variation in transmission time of packets.