|  |
| --- |
|  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SOFTWARE REQUIREMENT SPECIFICATIONs  **Global Optimization using Meta-Heuristics** | |  |  | | --- | --- | | **Faiza Shanawar** | **15140070** | | **Mohsin Qamar** | **15140104** | | **Haider Ali** | **15140101** | | **Usama Imran** | **15140098** | |

Supervised by:

**Syed Qamar Askari**

# Document Information

|  |  |
| --- | --- |
| User |  |
| Project | Meta-Heuristics for Global Optimization |
| Document Version | 1.0 |
| Document ID | AR-02 |
| Identifier | n/a |
| Status | Draft |
| Authors(s) | Mohsin Khan |
| Approver(s) |  |
| Issue Date |  |

# Jargons

|  |  |
| --- | --- |
| Optimization | The Selection of best set, unit in some set of available alternatives |
| Bench-mark function | The reliability, efficiency and validation of algorithms are tested through the benchmark functions. |
| Meta-Heuristics | A Heuristic designed that may or may not provide a sufficient best solution to an optimization problem |
| Evolutionary Algorithms | A nature inspired population based meta-heuristic algorithm |
| Visualization | Technique for creating images, figures |
| Contours of Benchmarks | The outline display of the solution space of a bench mark function |
| Standard Deviation | measure that is used to quantify the amount of variation or dispersion of a set of data values |
| Convergence Curves | Relationship between grid interval and analysis accuracy |
| Trajectory | The path and the curve followed by the algorithm |
|  |  |

Contents

[Document Information 1](#_Toc5703185)

[Jargons 1](#_Toc5703186)

[1. Introduction: 3](#_Toc5703187)

[1.1. Purpose: 3](#_Toc5703188)

[1.2. Document Conventions: 3](#_Toc5703189)

[2. Overall Description: 3](#_Toc5703190)

[2.1. Project Scope: 3](#_Toc5703191)

[2.2. User classes and characteristics: 4](#_Toc5703192)

[3. Operating environment: 4](#_Toc5703193)

[4. System Features: 4](#_Toc5703194)

[4.1.1. Algorithms Selection: 4](#_Toc5703195)

[4.1.2. Description: 4](#_Toc5703196)

[4.1.3. Functional Requirements: 4](#_Toc5703197)

[4.2.1. Benchmark Functions Selection: 5](#_Toc5703198)

[4.2.2. Description: 5](#_Toc5703199)

[4.3.1. Visual Result: 5](#_Toc5703200)

[4.3.2. Description: 5](#_Toc5703201)

[Product Functions: 6](#_Toc5703202)

[User Classes: 6](#_Toc5703203)

[Assumptions and Dependences: 6](#_Toc5703204)

[Assumptions: 6](#_Toc5703205)

[Dependencies: 6](#_Toc5703206)

[Constrains: 7](#_Toc5703207)

[Non-Functional Requirements: 7](#_Toc5703208)

[Usability factor: 7](#_Toc5703209)

[Learning outcomes: 7](#_Toc5703210)

# Introduction:

This is Software Requirements Specification Document for **Hikaku 1.0**. This Application would be used to compare optimization algorithms that could help the researchers to get the results of their applied algorithms.

## Purpose:

This SRS describe the functional and non-functional requirement for our application and it may include a set of use cases that describe user interactions that the software must provide. This document is intended to be used by developers only.

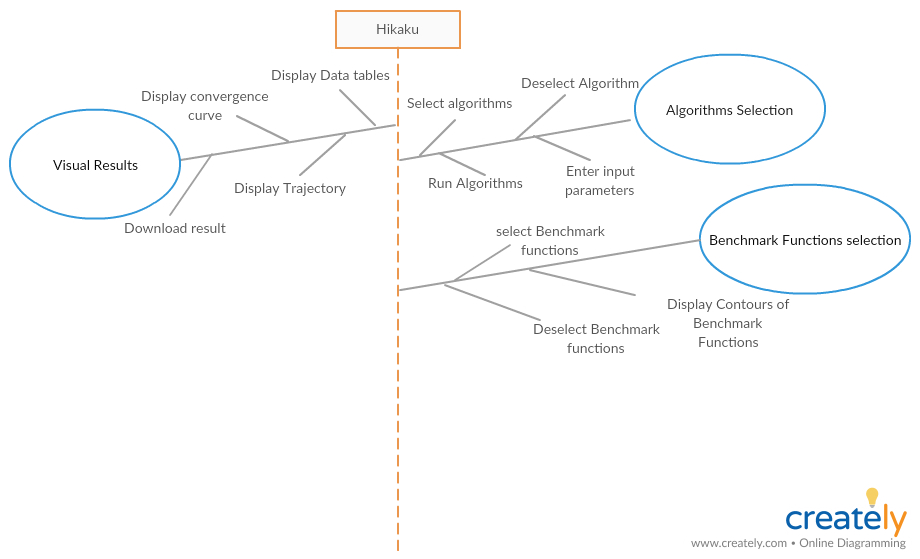
## Document Conventions:

|  |  |
| --- | --- |
| Functional Requirements | Hierarchical numbering convention is used while writing functional requirements. Functional Requirements with ‘\*’ are further elaborated by use cases. |

# Overall Description:

## Project Scope:

This application is intended to be used just for presenting research results. Therefore, is scope is limited.



## User classes and characteristics:

Intended users are:

1. Researchers
2. Teachers
3. Students

# Operating environment:

# System Features:

## Algorithms Selection:

## Description:

This feature shall enable user to select desired algorithm and run it on a specific Evaluation Function and then user can visualize results in the form of graphs

## Functional Requirements:

|  |  |
| --- | --- |
| Number | **Functional Requirements** |
| 4..1.3.1 | The system shall be able to provide the facility to select the Algorithm which needs to be run in a combo box. |
| 4.1.3.2 | The system shall be able to provide the facility to dis-select the Algorithm. |
| 4.1.3.3 | The system shall be able to provide the facility to enter inputs parameters which occurs in a popup after the algorithm is selected. |
| 4.1.3.4 | The system shall be able to provide the facility to run algorithms |

## 4.2.1. Benchmark Functions Selection:

## 4.2.2. Description:

This feature shall enable the user to select desired benchmark functions and the algorithms run and gives results for each benchmark function. The results will be displayed separately in form of graphs for each benchmark functions.

|  |  |
| --- | --- |
| Number | **Functional Requirements** |
| 4.2.3.1 | The system shall be able to select the benchmark function which needs to be run in a combo box. |
| 4.2.3.2 | The system shall be able to provide the facility to dis-select the benchmark function. |
| 4.2.3.3 | The system shall be able to provide the facility to display the contours of the Benchmark functions. |

## 4.3.1. Visual Result:

## 4.3.2. Description:

This feature shall enable the user to the compare the results of selected algorithm by plotting the results of each benchmarks functions separately in a graph. User can be change the graph type in the option given graphs

|  |  |
| --- | --- |
| Number | **Functional Requirements** |
| 4.3.3.1 | The System shall be able to provide the facility to display data of comparison of algorithms in a form of tables. Tables contains the average, Mean, Standard Deviation of the result of algorithm. |
| 4.3.3.2 | The system shall be able to provide the facility to download the graph plot result in form of pdf. |
| 4.3.3.3 | The System shall be able to provide the facility to display data of comparison in the form of convergence curves. The data is displayed in a graph which the rate of convergence is ds |
| 4.3.3.4 | The System shall be able to provide the facility to display data of comparison in the form of trajectory. In a Graph the values of parameters against each iteration is displayed |

# Product Functions:

In this application, user should be able to compare and show results of different algorithms through tables and different graphs. So, the main features that are included in the application are:

1. Problem Selection
2. Bench mark Functions.
3. Algorithm selection
4. Bench Mark Visualization(contours)
5. Results visualization

Users will be able to select the problem (Bench Mark Functions and algorithms to compare and results will be displayed in tabular form and graphs, plots of bench mark functions (user can also see the contours of the benchmark functions) and output of convergence curve of algorithms, trajectory and search history graphs.

# User Classes:

Following will be the regular user of the application:

1. Researchers
2. Teachers
3. Students

# Assumptions and Dependences:

## Assumptions:

Application should run on the system which has minimum requirements as follow: -

* 1 GB Ram or More
* 1.9 GHz or More
* 512 MB VGA card or more
* Windows 7 or more

## Dependencies:

Application depends on the following factors:

* Java JRE must be installed.
* Java JDK must be installed

# Constrains:

Constrains of the application are:

* User shall be provided the system as a desktop app. No web interface is provided.
* User cannot upload an algorithm.
* User cannot upload benchmark functions.
* User cannot plot data of multiple benchmark functions in a single graph.

# Non-Functional Requirements:

## Usability factor:

1. For beginner users only 3 to 5 trials are required to get full command on the operations (Problem Selection, Algorithm Selection and Results Visualization) of this application.
2. For experts only 2 trials are required to get full command on the operations of this application.

# Learning outcomes:

Following will be the learning outcomes of this project:

* We will be able to model real-life problems as optimization problems.
* We will be able to choose a solution method appropriate to the characteristics of a given problem and obtain a solution.
* We will be able to implement optimization methods yields problems in numerical linear algebra.
* Understanding of Meta-Heuristics and Evolutionary Algorithms
* We will be able to Mathematical formulation a concept or inspiration
* We will be able to learn how to manage and divide work.
* We will able to use different tools like MATLAB.