Jaipur Engineering College and Research Centre

Department of Computer Science and Engineering

Project Proposal

Course: B.Tech Computer Science **Subject: Major Project**

Team Member (1): Shubham Jindal **Roll Number: 15EJCCS160**

Team Member (2): Shubham Dudeja **Roll Number: 15EJCCS158**

Project Title: INSTANT SEARCH

Introduction & Background:

Instant-Search result page on an e-commerce website This project also includes a simplified version of the implementation that includes a few less filtering options and common Searching Algorithms . Instant search engine is a software system that is designed to search for information on the E-commerce Website or any other Sites . The search results are generally presented in a line of results often referred to as search engine results pages (SERPs).

Search engines are basically a web based tool that enables the users to find information on the Websites. SERPs is simply an acronym for Search Engine Result Pages. This page lists all the results found out for the particular query/keyword. Its simple; you enter your keyword (the word that you want to search for) and the Search Engines return with their list of websites that will provide you your desired results.

Instant Search Engines use different mathematical algorithms for generating Search Results. Different Search Engines perceive different elements of a web page including page title, content, meta description and then come up with their results to rank on. Search Engine's algorithm are different.

Instant Search is Fast and One time reloading page Search Bar of any website and give desired result as user want.

Instant search engines provide customized results based on the user's activity history. This leads to an effect that has been called a filter bubble. The term describes a phenomenon in which websites use algorithms to selectively guess what information a user would like to see, based on information about the user. As a result, websites tend to show only information that agrees with the user's past viewpoint. This puts the user in a state of intellectual isolation without contrary information.

Aims and Objectives:

Insant Search is a hosted full-text, numerical, and faceted search engine capable of delivering real-time results from the first keystroke. Instant Search powerful API lets you quickly and seamlessly implement search within your websites and mobile applications and delivering relevant results in under 100ms anywhere in the Website.

Speed

The entire system that manages your data and search requests was built to ensure speed at every point in the processing workflow. For example, search requests always have priority over indexing operations, to guarantee an optimal experience for your end users.

Relevance

The primary goal of the engine is to find all items that match a query, then to order them from best to worst. We explain what relevance and ranking mean in many parts of our documentation, but the overriding principle is that Algolia uses a tie-breaking algorithm, using a variety of criteria, to weigh and compare matching items against each other. This ensures that the best matches appear at the top.

Transparency

The methodology and criteria that Algolia uses to find records and rank them is openly available for your scrutiny. Furthermore, most of it is configurable, making all engine defaults adaptable to your unique needs. This is what we mean by transparency: Algolia has made every effort to provide a window into its search algorithm. There are no secrets or complicated statistical formulas.

To help you build the best solution as quickly as possible, we provide a family of UI/UX libraries called InstantSearch. InstantSearch offers a full set of UI components that you can use on any web, mobile, or voice application. In minutes, you get a fully-functional search UI. The InstantSearch widgets can be used out-of-the-box with no further customization. They can fit into your existing app as is, but they’re also fully customizable to match your needs.

Technologies & Resources:

In this phase we will discuss the project requirements that are laid down such as hardware required to run the project, additional software required to run the project, what functionalities are required in the project, other non-functional features that must be present in the project.

1.1 HARDWARE REQUIREMENT

Hard-disk drive	20 GB Min 40 GB Recommended
RAM	512 MB Min 1 GB Recommended

1.2 SOFTWARE REQUIREMENT

Operating System	Windows , Linux , MacOs
Database	MS SQL or MongoDB
Technologies	ReactJs, NodeJs , Searching Algoritms
Programming Language	JavaScript , Jquery , AJAX
IDE	ATOM(An Open Source IDE)
Browser	Google Chrome 6 , Mozilla , IE

Method & Work Plan:

The dashboard is an interface for accessing our data and index configuration. It's been designed for both developers and non-developers.

For developers, we recommend using our API clients for access to Instant Search full feature set as well as more dynamic control. However, to quickly test a static data set, it's possible to do an initial import using the dashboard.

Our dashboard is especially useful for project managers and other non-technical users to easily tweak index configuration settings after the data has initially been pushed.

This page lets you manage our API credentials. we can see our Application IDs and API Keys such as the Admin Key, Search-Only Key, and the Monitoring Key.

All of these Keys are security-sensitive, as they give direct access to your data, settings, and analytics. To be safe, you might want to regenerate them from time to time, which you can do on this page, or using the API.

To easily bootstrap a working JavaScript app in seconds, we'll use the create-instantsearch-app command line tool.

We can see in src/app.js that you are given your Algolia credentials to **instantsearch()** and you are using three **InstantSearch widgets**:

- searchBox displays a nice looking Search Box for users to type queries in it
- hits displays the results from Instant Search, based on the query.
- pagination displays the pages to show more results.

We have a lot more widgets, you can discover them all in the widget showcase.

To make our search UI more efficient and practical for your users, we add some more widgets:

- a way to filter the store by brands
- a way to clear active filters

InstantSearch.js can be used either with a direct link in your webpage or with a packaging system . This method uses the built version of **JavaScript** from the jsDeliver CDN. We will then have

access to the instantsearch function in the global scope (window).If you have a JavaScript build system, you can install **InstantSearch.js** from npm.

The widget lifecycle and API

InstantSearch defines the widget lifecycle of the widgets in 4 steps:

- The configuration step, during which the initial search configuration is computed
- The init step, which happens before the first search
- The render step, which happens after each result from Instant Search
- The dispose step, which happens when you remove the widget or dispose the InstantSearch instance

Those steps translate directly into the widget API. Widgets are defined as plain JS objects with 7 methods:

- **getConfiguration** optional, used to returns the necessary subpart of the configuration, specific to this widget
- **init** optional, used to setup the widget (good place to first setup the initial DOM). Called before the first search.
- **render** optional, used to update the widget with the new information from the results. Called after each time results come back from Algolia
- **dispose** optional, used to remove the specific configuration which was specified in the getConfiguration method. Called when removing the widget or when InstantSearch disposes itself.
- **getWidgetState** optional, used to get the UI state of the widget. This UI state is the object used to create the URL with the routing system.
- **getWidgetSearchParameters** optional, used to get the SearchParamters of the widget. This SearchParamters is used to create the correct request from an URL with the routing system.

I agree to Guide the students

……..…….. **……**

Guide Signature ***Date***

Shailesh Arrawatia

Submitted to Supervisor:

……..…….. **……**

Supervisor Signature ***Supervisor Signature***

Dr. Neelam Chaplot **Ms. Manju Vyas**