Indian Journal of Computer Science and Technology
Volume2, Issue1 (January-April 2023), PP: 01-10.
www.indjcst.com **ISSN No: 2583-5300**

Model Aphid Predator Neuroptera (Insecta): Web-Based Digital Key Development from Dichotomous Key by Computer Programming

Sourish Dutta[1], Rubia Mondal[2], Sayak Ganguli[3], Rayan Das[4], Santi Ranjan Dey[5]
[1,2,5] Department of Zoology, Rammohan College, Kolkata, West Bengal, India.
[3]Post Graduate Department of Biotechnology, Saint Xavier's College, Kolkata, India.
[4] Department of Zoology, Asutosh College, Kolkata, India.

Abstract: *The bulk of scientific problems given by biologists could never be answered or even envisioned without reference to some sort of hierarchical arrangement of tax a, making classification the most fundamental component of all biological sciences. The majority of biologists may agree that the goal of taxonomy should be to identify the many components of biodiversity, especially considering the significance of accurate organism identification for bio security and the best crop management practises. However, field experience in taxonomy and application is disappearing. As a result, it is essential to find any pests or other creatures. Since adopting new taxonomic technologies in 2011, the USDA Animal and Plant Health Inspection Service, Plant Protection and Quarantine (APHIS-PPQ) has created an incredible array of identification tools, some of which are gaining enormous popularity. There isn't yet a database or programme in India that is suited for this use and uses Indian algorithms. We have chosen the predator group of aphids as our model organism in order to create our own "Digital Key" experiment and identify insects that can control aphids as biocontrol agents. This is because the Neuroptera (Insecta) of West Bengal, India is region-specific database reliant (Homoptera: Aphididae). As a result, we made an experimental website called www.lacewingsofwestbengal.in and a "Digital Key" of "Families" of a particular group of Neuropterans.*

Keywords: *Aphid-predator, Hemerobiidae, Chrysopidae, Dilaridae, Coniopterygidae, Digital key*

I.INTRODUCTION

Most scientific problems posed by biologists could never be defined or even envisaged without reference to some sort of hierarchical organisation of taxa, making classification the most fundamental of the biological disciplines (Rieppel, 1992, 2004). But the question of the future of taxonomy—or even if it has a future—has started to crop up frequently in both specialised and general scientific publications. Taxonomy shouldn't serve as the "handmaiden" of other biological disciplines, it has been maintained (Wheeler 2008). On the other hand, the majority of biologists might agree that taxonomy's goal should be to identify the elements of Taxonomic and practical field knowledge, however, are both disappearing. As a result, it is critical to identify any pests or other organisms. After adopting new taxonomic technology in 2011, USDA Animal and Plant Health Inspection Service, Plant Protection and Quarantine (APHIS-PPQ) has created an astounding array of identification tools, several of which are gaining enormous popularity. The method is the creation of digital and beta keys. The algorithm has been created and registered for patent. 130 countries presently utilise this commercial software (LUCID BUILDER VERSION 3.5). India is still working on its own software. **As it is region specific database dependent, we have to develop our own algorithm.**

Members of the superorder Neuropterida (orders Neuroptera, Megaloptera, Raphidioptera, and Glosselytrodea) have access to a range of digital data online, including lacewings, antlions, fish flies, and snake flies. One can easily retrieve this information via the digital interface of Texas A&M University's Lacewing Digital Library (LDL). Since 2010, visitors have had access to a variety of online resources through the LDL portal (http://lacewing.tamu.edu), which provides information on neuropterid insects, global neuropterists, and related subjects. The Neuroptera or any other class of insects have yet to have digitised documentation produced in India.

The demand for taxonomists has reached previously unheard-of heights due to the practically continual appearance of new diseases and pests in agricultural areas. For experts in insects, identifying new, existing, and possibly invasive pest species is a daily problem. It is essential to appropriately identify any insects before taking any further action. In the past, end users have recognised insects using a range of techniques, including matching (type specimens), dichotomous keys, route keys, matrices and multiple entry keys (computer assisted), tabular keys (taxa vs. character states), and punch card keys. Recently, the use of computer-based taxonomic applications that include taxonomic keys, resources, and tools has increased. On the other hand, alpha taxonomy has advanced down the systematic branches. Most biological sciences require the ability to recognize species, particularly invasive ones. It does play a bigger part in pest management, though (Odeh et al, 2015, Sharma et al, 2019).

The group of neuroptera is crucial as a biological control agent. Neuropterans are currently being sold commercially as controlling agents by 14 Canadian, 33 Mexican, and 95 American firms (http://www.cdpr.ca.gov/docs/pestmgt/ip). Additionally, our state has a neuropteran resource. More than 35 different species of Neuroptera associated with aphids have been collected and identified by us in West Bengal. So, for the "Family," we created a model beta digital key.

II. MATERIALS

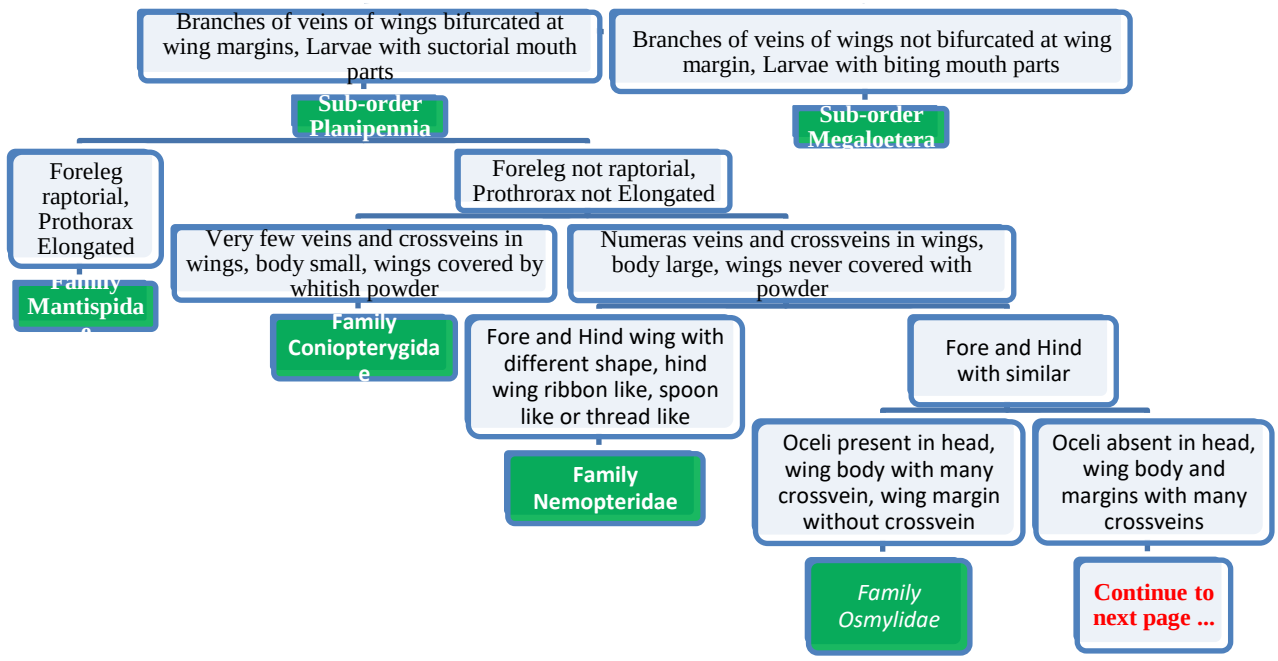
Key to the Aphidophagous Families of West Bengal of the Suborder Planipennia

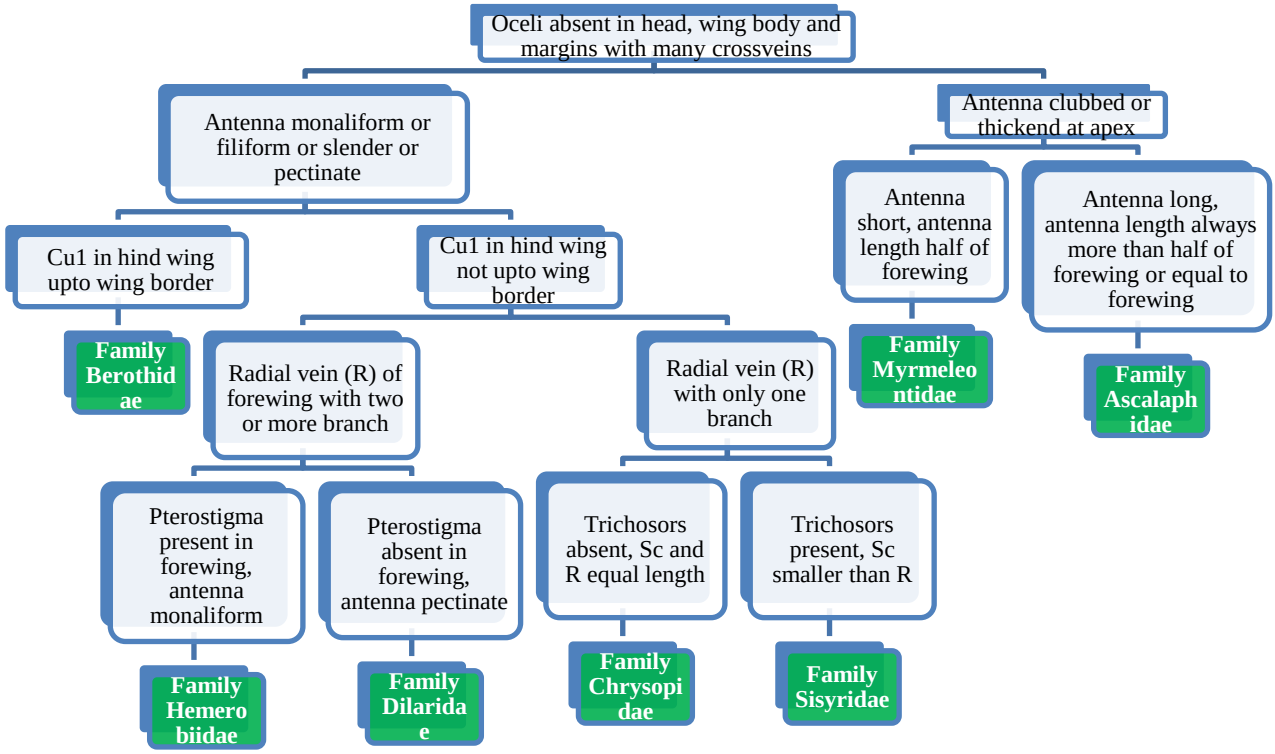
We have collected Neuropteran specimens from different regions of West Bengal and analyzed their morphology for

identification. From the identified specimens a family key has been constructed, which is as follows:

1. Fore leg raptorial with strongly thickned femur;prothor.....Mantispidae.
- Fore leg not raptorial; prothorax not elogated.....2.
2. Very few veins and crossveins in wings; body small; wings covered by whitish powdery exudation; venation without terminal twiggings.....Coniopterygidae.
- numerous veins and crossveins in wings; body large, wings never covered with such powdery exudation; wing venation with terminal twiggings.....3.
3. Fore and hind wing different shaped, hind wing ribbon like, spoon like or thread like; mouth produced into a short Beak.....Nemopteridae.
- Fore and hind wing similar; mouth conical without any beak.....4.
veinlets.....Osmylidae.
4. Ocelli present; discal area of wings with many crossveins, marginal area without crossveins but with many forked
- Ocelli absent; crossvein arrangement of wings not as above .....5.
5. Antennae monaliform or filiform, slender, pectinate, neither clubbed nor with thickened at apex.....6.
- Antennae cylindrical and always with clubbed or thickened at apex.....10.
6. Cu1 in hind wings running for a long distance close to hind border; costa and sometimes other parts of the females with seed like scales.....Berothidae.
- Cu1 in hind wing not as above; costa and other parts of the females without seed like scales .....7.
7. R of fore wing with two or more branches; costal veinlet complex.....8.
- R of fore wing with only one branch; costal veinlet simple.....9.
8. Pteostigma present; in forewing, Sc connected to R1 by crossveins; antennae monaliform.....Hemerobiidae.
- Pterostigma absent; in forewing, Sc not connected to R1; male antennae pectinate.....Dilaridae.
9. Trichosors absent; Rs with zigzag appearance; Sc and R usually equal in length.....Chrysopidae.
- Trichosors present; Rs without zigzag appearance; Sc usually smaller than R.....Sisyridae.
10. Antennae short, less than half of the fore wing, weakly clubbed or flattened towards apex; hypostigmatic cell elongated.....Myrmeleontidae.
- Antennae long, always more than half or as long as the fore wing, shortly clavate towards apex, hypostigmatic cell not elongated.....Ascalaphidae.

The Flowchart that is Converted into Digital Key





III.METHODS

The above dichotomous key and flow chart has been converted in to digital key by using the following "proper programme"

JQuery & Ajax

It is a JavaScript (JS) library that is open-source and was created with the motto "Write Less, Do More." John Resig created the technology, which he demonstrated in 2006 at the Bar Camp NYC conference. It was a development in front end development methodology where developers merely needed to call their method rather than repeatedly rewriting each job block.

jQuery syntax basically follows the same pattern as a code snippet for each statement. A method or function is applied to an event after a selected element from the HTML DOM is chosen: $(selector).method_Or_Function(); The majority of jQuery's statements apply various functions and methods to particular HTML elements and their attributes. Depending on the selection and the action, which is the applied method or function, this technique is easily modifiable. Jquery technology makes it simple to manage and manipulate DOM items. Currently jQuery is used on about 20% of all sites on the internet and has an extensive adoption rate from companies large and small. JQuery has been used deeply by companies like Twitter, Kick starter, Uber, Hoot suite, Angel List and others.

There are 3 main tasks perform using jquery mostly on front end -

- *DOM traversal and manipulation*

```
$( "button.continue" ).html( "The content following…" )
```

The above code will find/select all 'button' element with class 'continue' and change their html content to the given string passed into html().

- *Event Handling*

```
var hb = $( "#welcome-message" );
$( "#bt button" ).on( "click", function( event ) {
        hb.show();
});
```

The above code will display a hidden box with welcome message whenever 'click' event fired on any button in group #bt button.

- *Ajax*

It is basically used to call server side script in order to get some result/response from the server query or process without reloading the whole content. Sample code is given as -

```
$.ajax({
        url: "Some api or url",
        data: {
                somevar: value of var
        },
```

```
        success: function( result ) {
            $( "#some_id" ).html( "content to display as result from 'result'" );
        }
});
```

In the current scope, we have used these two technologies (jquery & ajax) almost everywhere for front end development. Representation of all tables and key generation interface are also developed with these techs in order to maintain the flexibility for a variety of devices, such as – desktop, laptop, tab, mobiles etc. It also provides the state of the art look and feels for the entire portal.

PHP & CI4

Very popular web based server side scripting technology termed as PHP - Hypertext Preprocessor, mainly contains the backend and all business logic for the portal where CI4 is a framework under PHP with advanced OOP concept and a bit distributed architechture.

Some of the important task in current scope -

- Dealing with database
- Converts algorithms to code
- Some exception handling
- Distributing structure with CI4
- Routing urls using CI4 utility

According to some estimates, 78.1% of all websites—including popular ones like Face book and Wikipedia—use PHP. A script is a collection of commands or instructions intended to be carried out on other programmes or applications. It is an open-source server-side scripting language that is commonly used in web development. Currently, there are primarily two categories of scripting languages: one is designed for the client-side, or front-end, and the other is for the server-side, or back-end.

Because PHP is a server-side scripting language, the server, not the client's computer, actually runs the script's instructions. A variety of these scripts have been used to do tasks including providing data on demand, channelling requests, and organising the information in a database on the server.

When a web server receives a script, it will process the request and send output to a web browser in an HTML format. A web server database stores the information so other users can't access the data and source code. With scripts, users can perform operations like customizing a site, making dynamic changes to website content, and accessing a database. Users also often use a scripting language to develop web applications.

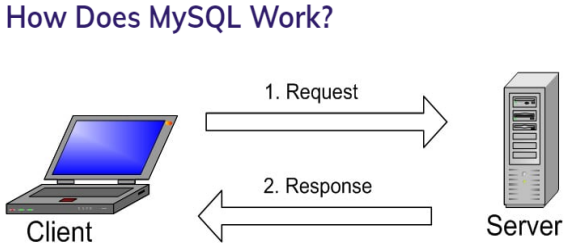
MySql

RDBMS, or Relational Data Base Management System, is a very common database in industry. In 1994, a Swedish corporation by the name of MySQL AB created MySQL. When Sun Microsystems acquired MySQL AB in 2008, they did so as the sole owner. Sun Microsystems was purchased by US technology behemoth Oracle in 2010, and since then Oracle has effectively owned MySQL.

We can call a database is collection of structured data. A database is a place where data is stored and organized in a well defined procedure. The word "relational" means that the data stored in the dataset is organized as tables. Every table relates in some ways.

Computers that install and run RDBMS software are typically called clients. Whenever they need to access data, they connect to the RDBMS server. That's the "client-server" part.

Obviously MySQL is one of many RDBMS software options among Ms-SQL, Oracle, Cybase etc. Many big web applications like Face book, Twitter, YouTube, Google, and Yahoo! all use MySQL for data storage purposes. Even though it was initially created for limited usage, it is now compatible with many important computing platforms like Linux, macOS, Microsoft Windows, and Ubuntu.



We planned to use MySql as our data storage for this project as it is open source and flexible for use. All records are securely stored in separate tables under specific database in mysql. Certain dedicated scripts were written which are responsible for data transfer and transformation.

Html & CSS

Is a web platform recommended by W3C for any publication? specifies how each page and part is organised. The most recent version of HTML, version 5, was used to design every structure in the present scope.

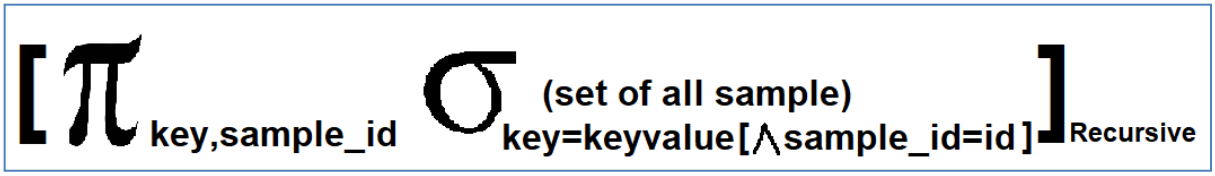
Cascading Style Sheet, or CSS, is used for various style purposes. Initially, we had intended to build from scratch, but we ultimately decided to use the BS5 (Bootstrap 5) framework because it allowed for quicker development, a nicer look and feel, and the availability of the majority of user-friendly components that had been tested for a long time.

The Programme

The following are the logic/algorithm implemented in client-server format like a simple interrogation or conversation. Each characteristics are assigned with a key-value to be represented in the characteristic database for species and genus separately.

Server prompts a specific characteristic with options to acquire Client's knowledge on sample with him/her. Depending on client's feedback or choice, it ask module to find suitable branching to follow next thus reaches to conclusion or final diagnosis.

Unique search key is used individually for family ranging from A – Z with the following basic relational algebraic query pattern =>



Front End (Client) part

```
$(document).ready(function(){
        $("#nfamily").hide();
        $("#ngenus").hide();
        $("#btnfamily").click(function(){
                $.ajax({
                    url:"digikey/family",
                    type:"post",
                    data:"dataSymbol=0",
                    dataType: 'json',
                    error:function(){alert("Something went wrong!!!");},
                    success:function(response){
                            var s='';
                            $.each(response, function(index, element) {
                                    s=s+'<div class="card">'+
                                    '<button class="btn btn-light">'+
                                    '<div class="card-body">'+
                                    '<h5 class="card-title" id="nfamilyh5"
                                    style="display:none"><code>'+element.symbol+'</code></h5>'+
                                    '<p class="card-text" id="nfamilyp">'+element.key+'</p>'+
                                    '</div>'+'</button>'+'</div>'+'<br/>';
                                    });
                                    $('#nfamilycardbody').html(s);
                            }//end success
                });
                $("#nfamily").toggle(500);
                $("#ngenus").hide();
        });
        $("#btngenus").click(function(){
                $.ajax({
                    url:"<?=base_url('digikey/genus')?>",
                    type:"post",
                    data:"dataSymbol=0",
                    dataType: 'json',
                    error:function(){alert("Something went wrong!!!");},
                    success:function(response){
                            var s='';
                            $.each(response, function(index, element) {
                                    s=s+'<div class="card">'+
                                    '<button class="btn btn-light">'+
                                    '<div class="card-body">'+'<h5 class="card-title" id="ngenush5"
```

```
style="display:none"><code>'+element.symbol+'</code></h5>'+'<p class="card-text"
id="ngenusp">'+element.key+'</p>'+'</div>'+'</button>'+'</div>'+'<br/>';
                                });
                                $('#ngenuscardbody').html(s);
}//end success
        });
        $("#ngenus").toggle(500);
        $("#nfamily").hide();
});
$(document).on('click','#nfamilycardbody>.card',function(e){
        var dataSymbol=$(this).find("#nfamilyh5>code").html();
        var dataStr="dataSymbol="+dataSymbol;
        if(dataSymbol!='X'){
                $.ajax({
                url:"digikey/family",
                type:"post",
                data:dataStr,
                dataType: 'json',
                error:function(){alert("Something went wrong!!!");},
                success:function(response){
                        var s='';
                        $.each(response, function(index, element) {
                                var last=element.symbol[element.symbol.length -1];
                                if(last=='X'){
                                        s=s+'<div class="card">'+'<button class="btn btn-danger">'+
'<div class="card-body">'+ '<h5 class="card-title" id="nfamilyh5"  style="display:none"> <code>'+
element.symbol+'</code></h5>'+'<p class="card-text" id="nfamilyp">'+element.key+'</p>'+
'</div>'+'</button>'+'</div>'+'<br/>';}
                                else if(last=='x'){
                                        s=s+'<div class="card">'+'<button class="btn btn-success">'+
'<div class="card-body">'+'<h5 class="card-title" id="nfamilyh5" style="display:none"> <code>'+
element.symbol+'</code></h5>'+'<p class="card-text" id="nfamilyp">'+element.key+'</p>'+
'</div>'+'</button>'+'</div>'+'<br/>';}
                                else{
                                        s=s+'<div class="card">'+'<button class="btn btn-light">'+
'<div class="card-body">'+'<h5 class="card-title" id="nfamilyh5" style="display:none"> <code>'+
element.symbol+'</code></h5>'+'<p class="card-text" id="nfamilyp">'+element.key+'</p>'+
'</div>'+'</button>'+'</div>'+'<br/>';}
                                });
                                $('#nfamilycardbody').html(s);
                        }//end success
                });
        }
});
```

Back End (Server) part

```
family(){
if(isset($_POST) && $_POST['dataSymbol']!=''){
$dataSymbol=$_POST['dataSymbol'];
$data=array();
if(substr($dataSymbol,-1)=='x'){
$data=array(
array('symbol'=>'X','key'=>'You are at end of a branch/family')
);}else if($dataSymbol=='0'){
$data=array(array('symbol'=>'A','key'=>'Branches of veins of wings bifurcated at wing margins, Larvae with suctorial mouth
parts'),array('symbol'=>'B','key'=>'Branches of veins of wings not bifurcated at wing margin, Larvae with biting mouth
parts'));}else if($dataSymbol=='A'){$data=array(
array('symbol'=>'Ay','key'=>'Sub-order Planipennia'));}else if($dataSymbol=='Ay'){$data=array(
array('symbol'=>'C','key'=>'Foreleg raptorial, Prothorax Elongated'),array('symbol'=>'D', 'key'=>'Foreleg not raptorial,
Prothrorax not Elongated'));}else if($dataSymbol=='C'){$data=array(
        array('symbol'=>'Cx','key'=>'Family Mantispidae'));}else if($dataSymbol=='D'){$data=array(array ('symbol'=>'E',
'key'=> 'Very few veins and crossveins in wings, body small, wings covered by whitish powder'), array ('symbol'=>'F', 'key'=>
'Numeras veinsand crossveins in wings, body large, wings never covered with powder'));}else
```

```
if($dataSymbol=='E'){$data=array(array('symbol'=>'Ex','key'=>'Family Coniopterygidae'));}else
if($dataSymbol=='F'){$data=array(array('symbol'=>'G','key'=>'Fore and Hind wing with different shape, hind wing ribbon like,
spoon like or thread like'),array('symbol'=>'H', 'key'=>'Fore and Hind with similar'));}else
if($dataSymbol=='G'){$data=array(array ('symbol'=>'Gx','key'=>'Family Nemopteridae') );}else if($dataSymbol=='H')
{$data=array(
array('symbol'=>'I','key'=>'Oceli present in head, wing body with many crossvein, wing margin without crossvein'),
array('symbol'=>'J','key'=>'Oceli absent in head, wing body and margins with many crossveins'));}else
if($dataSymbol=='I'){$data=array(array('symbol'=>'Ix','key'=>'Family Osmylidae') );}else
if($dataSymbol=='J'){$data=array(array('symbol'=>'K','key'=>'Antenna monaliform or filiform or slender or
pectinate'),array('symbol'=>'L','key'=>'Antenna clubbed or thickend at apex')
);}else if($dataSymbol=='K'){$data=array(array('symbol'=>'M','key'=>'Cu1 in hind wing upto wing
border'),array('symbol'=>'N','key'=>'Cu1 in hind wing not upto wing border') );}else
if($dataSymbol=='M'){$data=array(array('symbol'=>'Mx','key'=>'Family Berothidae'));} else if($dataSymbol=='N')
{$data=array(array('symbol'=>'Q','key'=>'Radial vein (R) of forewing with two or more branch'),
        array('symbol'=>'R','key'=>'Radial vein (R) with only one branch')
);}else if($dataSymbol=='Q'){$data=array(array('symbol'=>'S','key'=>'Pterostigma present in forewing, antenna monaliform'),
array('symbol'=>'T','key'=>'Pterostigma absent in forewing, antenna pectinate')
);}else if($dataSymbol=='S'){$data=array(array('symbol'=>'Sx','key'=>'Family Hemerobiidae')
);}else if($dataSymbol=='T'){$data=array(                    array('symbol'=>'Tx','key'=>'Family Dilaridae')
);}else if($dataSymbol=='R'){$data=array(array('symbol'=>'U','key'=>'Trichosors absent, Sc and R equal
length'),array('symbol'=>'V','key'=>'Trichosors present, Sc smaller than R')        );
}else if($dataSymbol=='U'){$data=array(array('symbol'=>'Ux','key'=>'Family Chrysopidae'));}else
if($dataSymbol=='V'){$data=array(array('symbol'=>'Vx','key'=>'Family Sisyridae'));}else if ($dataSymbol
=='L'){$data=array(array('symbol'=>'O','key'=>'Antenna short, antenna length half of forewing'),
array('symbol'=>'P','key'=>'Antenna long, antenna length always more than half of forewing or equal to forewing'));}else
if($dataSymbol=='O'){$data=array(array('symbol'=>'Ox','key'=>'Family Myrmeleontidae'));}else
if($dataSymbol=='P'){$data=array(array('symbol'=>'Px','key'=>'Family
Ascalaphidae'));}else if($dataSymbol=='B'){$data=array(array('symbol'=>'Bx','key'=>'Sub-order Megaloetera'));
}return json_encode($data);}echo 'You are not authorized here...';return false; }
```

Website Performance Metrics

In review, the six most important website performance metrics has been measured, that includes:

a. The speed index, also known as the **Above the fold rendering time**;
b. Finally interactive time;
c. Time to first byte;
d. Server response time;
e. DOM loading and processing; and
f. Webpage fully loaded time.

a. Above the fold rendering

This indicator -- also known as the speed index -- represents how quickly viewers can view the part of the webpage that displays before they scroll down. It's this pre-scroll area of the page that is referred to as above the fold -- terminology rooted in paper newspapers that fold up. This indicator shows the first instance in which a user finds your page beneficial. The end user will believe that the entire homepage has loaded, even if JavaScript events or below-the-fold items are still being processed or rendered on your page. But when keeping an eye on it, there are two things to consider.

I. This measure depends on the size of the user's view port. A 1080p widescreen display has a different quantity of content that renders above the fold than a 4K ultra-high definition display. Before implement this webpage performance metric, agree on a standard port size to reliably analyze the view.
II. This statistic likewise lacks a background event or browser function for its calculation. IT teams need to utilise a technology that records the invocation of a webpage and then analyse the time it takes for the page to completely render to acquire the metric's value. It can be a time-consuming operation, but above the fold rendering is still worthwhile to track.

b. Webpage finally interactive time

For pages that display content, above-the-fold rendering is very crucial. Simply showing information above the fold isn't sufficient for web-pages that contain input fields or clickable buttons, though. The inability of a user to complete a form field or click a submit button due to an incomplete script is the worst possible user experience a user may have on a website. The last interactive indicator counts the number of input fields and clickable elements that are operational.

On sites based on serial peripheral interfaces and created with Angular or Ember, this statistic is crucial. These frameworks allow page transitions to take place within the browser without starting a fresh request response cycle. For sites built with modern, client-side rendering frameworks, the finally interactive metric acts as a measure of when the user can navigate throughout the site.

c. Time to first byte

The time to first byte (TTFB) measures how long it takes for a client's browser to receive a response from the server after sending a request. Generally speaking, a desirable TTFB value is 200 milliseconds. TTFB is fully dependent on an organisation's hardware and network architecture, as opposed to how architects and developers created the webpage, like other webpage performance indicators. The TTFB may be impacted by a slow network, servers that are overloaded with a heavy load, and poorly planned network architecture. The TTFB number may be pushed into the one- to two-second range by any of these problems. As a result, it can be difficult for organizations to meet a three or four second target for a full page load.

d. Server response time

This statistic measures the difference between the TTFB and the moment the last bit of data is sent to the browser by the server. All provided images, all downloaded CSS files, and any postponed JavaScript files are all included. This measure is affected by infrastructure since fast servers are facilitated by powerful servers, the utilisation of content delivery networks, and an in-memory cache of static material.

Using techniques like page weight reduction, merging numerous JavaScript files into one, and using inline CSS rather than referring to many external style sheets, website developers can also contribute to the improvement of this measure. However, just because a user downloads every single one of these files doesn't guarantee that they have all been handled. Apply each relevant document object model (DOM) selector when the CSS files have downloaded. The webpage's onReady events must still be handled after downloading a JavaScript file. The completion of client-side rendering of the page may take several seconds after all materials have been retrieved from the server.

e. DOM loading and processing

This DOM processing metric calculates the time needed by the browser to parse the downloaded HTML file, create the corresponding document object model, create and initialise the CSS object model, and then trigger the DOM Content Loaded event.

This measure is greatly impacted by the size and structure of the webpage. It can easily take three or four seconds to parse a huge HTML file with over 1,000 DOM elements, and DOM processing can be slowed down by a CSS file with over 1,000 items. Just consider it: If there were 1,000 DOM items and 1,000 CSS declarations, the browser would have to perform 1,000,000 Boolean CSS selectors. These file sizes may cause slower mobile devices and laptops to stutter. The easiest approach for businesses to increase the speed at which a browser loads and parses the DOM is to lower the size of the DOM and the amount of CSS styles in use.

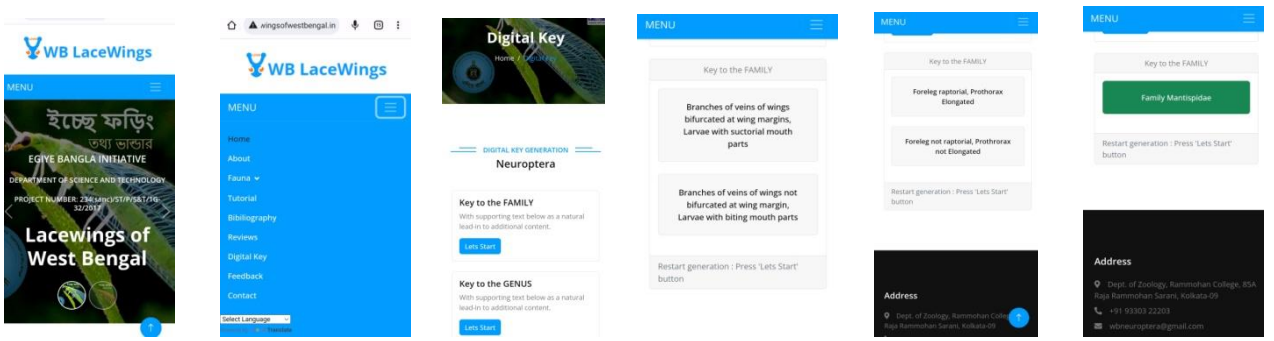
f. Webpage fully loaded time

The most popular statistic used to assess a webpage's performance is the amount of time it takes to fully load, often known as page load time. The statistic calculates how long it takes from the browser's request for the page until all associated resources have downloaded, all pictures have rendered, all CSS selectors have been used, and all onReady JavaScript events have triggered.

The fully loaded statistic overlooks some of the subtleties of the other metrics for webpage speed, such as the time it takes for a webpage to become interactive or to render content above the fold. These alternative measures are more accurate gauges of customer satisfaction. However, the page load time is the one that is most frequently mentioned when page performance is an issue because it is simple to quantify and understand. An organisation may detect any impending performance issues, take corrective action before they become problems, and guarantee that users consistently have a great experience on the website by continuously monitoring these crucial webpage performance metrics, especially on the website's main landing page.

Result I

The result is a web based platform www.lacewingsofwestbengal.in hosting the digital key to the "Family" of Neuropteran Insects. A sequence of digital key is shown below.



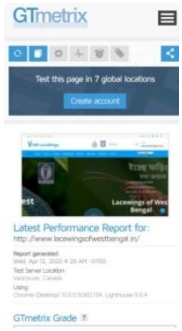
A B C D E F

A. Home page, www.lacewingsofwestbengal.in
B. Menu showing "Digital Key"
C. Digital Key Tab of "Family"

D. Inside "Family Key" 1st series character (Select 1st one)
E. Inside "Family Key" 2nd Series character (Select 1st one)
F. Conclusion "Family Mantispidae"

Result II

A



B

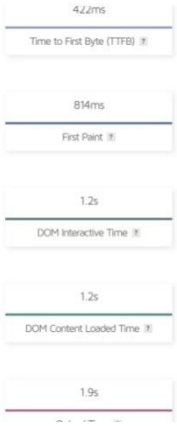
C

D

E

F

G



H

I

A. Website uploaded in GTmetrix
B. Website opened in GTmetrix
C. Overall performance of website
D. Visualization metrix
E. Speed Visualization
F. Page details of website
G. Server Response time
H. Website full loading time
I. DOM loading metrix

IV.DISCUSSION

A website platform has been made with the intention of hosting the "Digital Key to the Family." Anyone can translate the website's material into any language by using the "Selecting the Language" page. The "Home" provides general descriptions of the four (four) aphidophagous (aphid predator) families of the order Neuroptera (Insecta) for the benefit of laypeople. The "Fauna" section includes the Genus and Species of Aphid Predator Neuroptera discovered in West Bengal, India. The "Digital Key of Family" has been developed using the "Species" discovered in West Bengal, India. The "Digital Key" allows users to choose specific morphological traits to identify the "Family" if they are interested in learning more about Neuropteran Taxonomy. The "Review" includes information for laypeople about the biology, evolution, and utility of aphid predator neuroptera. The works we consulted for this project are listed in the "Bibliography." The main drawback of the "Family Digital Key" is that, it does not contain "Backward KEY" programme. To move back, one has to start from the very beginning of the Key. As per "Performance Metrics" this platform could achieve only 75% of its target and achieved "C" grade.

V.ACKNOWLEDGMENT

The authors are indebted to "Department of Science & Technology and Biotechnology", Government of West Bengal, India (Project No. 234(Sanc)/ST/P/S&T/1-G32/2017)for funding. The authors are also indebted to Principal, Rammohan College for providing laboratory facilities.

REFERENCES

1. *Odeh, A., Aymen, A.E. and Awad, M.. Symmetric Key Generation Method using Digital Image. Journal of Computer Science. 2015; 12(2): 254-259.*
2. *Rieppel, 0..Homology and logical fallacy. J. Evolutionary Biology. 1992; 5: 701-715.*
3. *Rieppel, O..The language of systematics, and the philosophy of "total evidence." Systematics and Biodiversity. 2004; 2: 9-19.*
4. *Sharma, N., Colucci-Gray, L., Siddharthan, A., Comont, R. and Van der Wal, R.. Designing online species identification 10.7717/peerj. 2019;5965.*
5. *Wheeler, Q.D. The new taxonomy. The Systematics Association. Special Volume Series 76.CRC Press. Boca Raton, FL. 2008.*
6. *https://www. lacewing.tamu.edu*
7. *https://www.cdpr.ca.gov/docs/pestmgt/ip*
8. *https://www. lacewingofwestbengal.in*