An Overview of Educational Methodologies and Visual Programming Language Designs to Aid Programmers with Dyslexia

Sameer Mahajan[1], Adwait Patil[2], Nahush Kulkarni[3]

[1,2,3]*Student, Department of Computer Engineering, TEC, University of Mumbai, Mumbai, India*

Abstract - *Dyslexia is a neurobiological situation where a person has trouble comprehending text as the phonological component of memory which is a part of the working memory is not as efficient as that of a non-dyslexic. Phonological loop governs phonemes which are responsible for distinguishing one word from another and weak phonetic skills, in turn, disturb the readability of the text. Around 50-100 out of 1000 students have learning disabilities like Dyslexia. Thus, a part of students faces a lot of difficulties in coping with the traditional educational pattern. This survey aims to cover how and why Dyslexia develops in a student, what are the general difficulties faced by the dyslexic population in various aspects of life. Considering this, if some changes are made in the educational teaching methodology, it can reduce their cognitive load. Therefore, supporting visuospatial skills and in turn, improve their performance. The main focus of this survey is to improve the efficiency of dyslexic programmers. In order to achieve this, we have discussed various Visual Programming Languages (VPL) and their functionalities in bridging the gap between a dyslexic programmer and a textual programming language-based programmer. According to studies, the dyslexic population is right-brained and have good visuospatial skills [4] and thus they prefer VPL's over the traditional programming methods. We also discuss the possibilities of inclusion of VPL's in high-level programming which is the need of the hour in software development. This might generate more job opportunities for Dyslexic programmers.*

Index Terms: Dyslexia, Dyslexic Programming, Visual Programming Languages, Scratch, Flip, Fabrik

1. INTRODUCTION

Dyslexia is a common language based "hidden" disability. Of the people having some reading disabilities, 70-80% are plausible to some form of dyslexia. The common causes of Dyslexia are reading, writing, and spelling intricacies. Almost 5-10% of the population is dyslexic and this percentage can go as high as 17% [1][2]. Defining dyslexia is an ambiguous topic in the real world. For example, a non-expert person would define dyslexia as a reading problem, a teacher would say dyslexic students fail to learn irrespective of proper teaching due to reading incompetence. These definitions suggest that dyslexic students may have a reading problem but studies suggest that they might face difficulties in general information processing. Dyslexia has more to do with memory and attention problems than literacy skills [2]. The most authoritative definition of dyslexia today is that of the International Dyslexia Association (IDA) based in the United States.

Dyslexia is characterized by difficulties with unerring or coherent word identification and poor attentiveness and awareness of spellings and sentences. The genesis of these challenges is neurobiological in nature. Other concomitant challenges may include lesser vocabulary, inefficacious reading, reduced working memory timespan, and comprehension intricacies. Phonological loop being a component of the working memory, scantiness in this component results in reduced word speech recognition. This in turn causes a lot of cognitive stress [3].

According to research, the Familial clustering of dyslexia has been well documented for almost 50 years. Various studies have suggested that locus predisposing to specific reading disability (dyslexia) resides on chromosome 6p23-p21.3. Testing 79 families with at least 2 affected siblings for any linkage between dyslexia and chromosomes, no evidence was found for linkage based on LOD score analysis or affected-sib-pair methods. Some evidence was found while using the affected-pedigree-member (APM) method. But those readings were not so significant when allele marker frequencies from parents were used. Results from a more robust SIMIBD method were also not significant when published or parental markers were used. Also, the AFBAC method did not show association with any marker. Hence concluding that the linkage between dyslexia and chromosome 6p cannot be confirmed via previous reports [22].

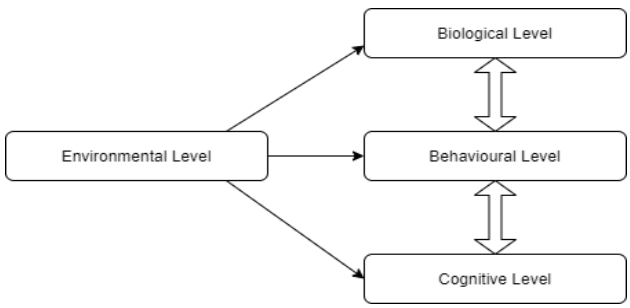

Fig -1: Levels of Dyslexia

Dyslexic challenges can be compartmentalized in four spectrums. Environmental Level drives factors like socio-economic and instructional factors. Biological Level checks brain and neurobiological functioning. Behavioral Level consists of reading comprehension and spelling problems. Cognitive Level governs mental processing and learning mechanisms [4].

- **Learning Procedure**

The memory is classified into two categories, the long-term memory, and the working memory. The long-term memory is responsible for storing information indefinitely. The

long-term memory is divided into two components Declarative Memory and Procedural Memory. Working Memory is managed by the Central Executive which governs two components, the Phonological Loop and Visuospatial Memory.

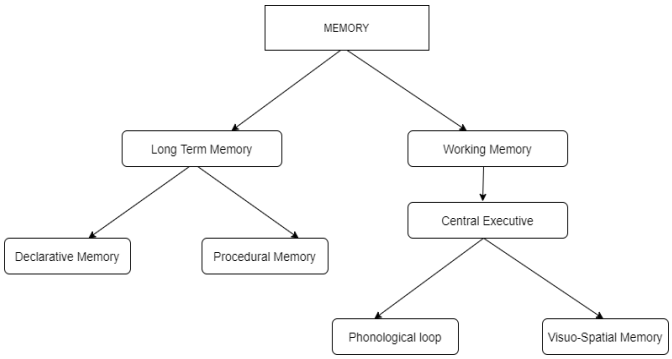

Fig -2: Memory Hierarchy

Declarative memory of the long-term memory is responsible for processing information related to facts, for example, Delhi is the capital of India, your Birthdate, etc. The Procedural Memory processes information of the "how to do" category, for example how to play a musical instrument, how to drive a car, etc. [2][5]. Working memory is a gateway to long term memory before the information is processed in long term memory it passes through the working memory [2]. Working memory has a central executive memory which is responsible for the monitoring and coordination of the Phonological Loop and the visuospatial sketchpad. The Phonological Loop is responsible for speech recognition, it processes speech-based information. The timeframe for speech recognition in the phonological loop of the working memory is around 1-2 seconds as we repeat the speech the information fits into the phonological store, for example remembering a phone number. The Visuospatial Sketchpad as the name suggests deals with the processing of visual and spatial information [6][2], for example, walking with people around the visuospatial sketchpad helps us in positioning the people around us so that we don't bump into them.

- **Reading Procedure**

Understanding the reading mechanism will take us closer to the difficulties the dyslexic population face. Reading is an activity in which word recognition and comprehension are essentially vital qualities. The comprehension needs to be automated which aids in a faster understanding of the sentence. Word recognition involves phonological processing to convert words into sound, extraction of information related to words, and understanding the usage of prefixes and suffixes [2].

The existing research suggests that the root cause of failure in word recognition for the dyslexic population is the problems in phonological information processing. As word recognition is the foundation of sentence comprehension dyslexic students find reading as onerous and labyrinthine. Thus, students face problems such as converting words into phonological units, slow speech, word recognition, difficulties in spellings, smaller vocabulary, articulation which involves the conversion of written words into sounds and then breaking them into phonological units [2]. Jeffries and Everitt 2004 suggest that the dyslexic population also has a shorter working memory time frame. Dyslexic students and adults face a lot of difficulties in concentrating for a long time due to less working memory time frame. Concentrating on a topic for long is a problem which the dyslexics face and thus they are predisposed to make mistakes. Some of the difficulties due to smaller working memory are difficulties with arithmetic and multiplication tables, prolong attentiveness, etc. [2].

2. CHALLENGES FACED BY DYSLEXICS

The amendments in policies such as the "Convention on the Rights of Persons" make it possible for people with dyslexia to become computer programmers [17]. A dyslexic programmer may underperform with traditional programming languages but with the help of visual programming languages, they can give their optimum performance. However, we need to understand their profile as programmers in order to come up with possible solutions. For this purpose, we need to understand the challenges faced by the dyslexic population in general.

Routine aspects of life where dyslexic people face challenges are:
1. Career
2. Reading, writing
3. Time management
4. Directions
5. Behavior

- **Career**

1. Trying to find a job/position where they (dyslexic programmers) are less exposed to interactions.
2. Conscious about sharing problems with co-workers, friends, or family.
3. Sequential tasks like meetings are extremely frustrating.
4. May have difficulty focusing on a single task at hand and might get diverted to multiple tasks easily.
5. May face difficulties passing standardized tests resulting in stunted growth in career.
6. May face high levels of anxiety. [18]

- **Reading, Writing**

1. Difficulty in reading unfamiliar fonts.
2. Conscious while reading out loud.
3. Has developed some tricks to remember spellings or homonyms (their, there, they're, etc.) but mostly misspells words, or misuses homonyms in unfamiliar conditions.
4. Takes more time reading to grasp the meaning.
5. Always uncertain about words, punctuations, and spellings while writing.

6. May jumble similar looking or sounding letters. [18]

- **Math, Time Management, Directions**

1. Excellent in understanding mathematical concepts but poor at presenting it on paper.
2. Difficulty in understanding multiple directions for E.g.: Go straight then take a right, follow towards north then take a left.
3. Problems in judging things that are easy for normal people, like judging where the ball will fall to catch it is hard for a dyslexic whereas it is a regular task for a normal person.
4. Easily loses track of time, or sometimes is too punctual. [18]

- **Behavioral**

1. They may become petulant when a lot of sequential tasks are assigned.
2. They may have low self-esteem.
3. They become conscious while working out of their comfort zone.
4. Either they are extremely disorderly or compulsively orderly. [18]

The following graph portrays a comparison between Dyslexic programmers and Non-dyslexic programmers based on various behavioral parameters.

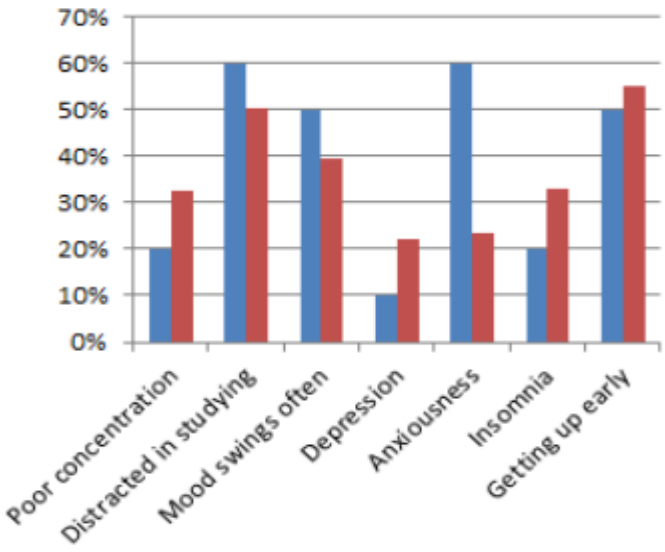

Chart -1: Behavioral Parameters

Chart 1 gives a graphical representation of a comparison between the number of dyslexic and non-dyslexic programmers who face the following behavioral problems [17].

1. Poor concentration only 20% of dyslexic programmers suffer from poor concentration as against 32.7% non-dyslexic programmers.
2. Distracted in studies, though most programmers concentrate during programming but get easily distracted while studying as 60% dyslexic and 50.3% non-dyslexic programmers face this problem.
3. Mood swings are a problem for half of the dyslexic programmers whereas only 39% non-dyslexic face it.
4. Depression is a minimal issue for dyslexia programmers are only 10% suffer depression as against 22.3% non-dyslexic programmers.
5. Anxiousness is surprisingly a major issue as 60% dyslexic programmers face it whereas only 23.4% of non-dyslexic programmers suffer from anxiety.
6. Only 20% of dyslexic programmers face insomnia, whereas 32% of non-dyslexic programmers face it.
7. Getting up early is a major issue for both dyslexic and non-dyslexic programmers as 50%, 55% of programmers respectively face this problem.

3. PROPOSED TEACHING METHODOLOGY

The dyslexic students are right-brained students, the right brain provides functions such as art-awareness, creativity, imagination, 3D forms, and a holistic thought process for a given task. The left-brain activities such as language, reasoning, number skills, writing, logic require a lot of practice for a dyslexic student to cope up with a normal student [7][8]. The teaching methods adopted by any classroom should bolster the process of learning via Visuals, in which the dyslexic students are good in a way to indemnify the weaker phonological skills [7]. Some changes in classroom teaching may boost the performance of a dyslexic student.

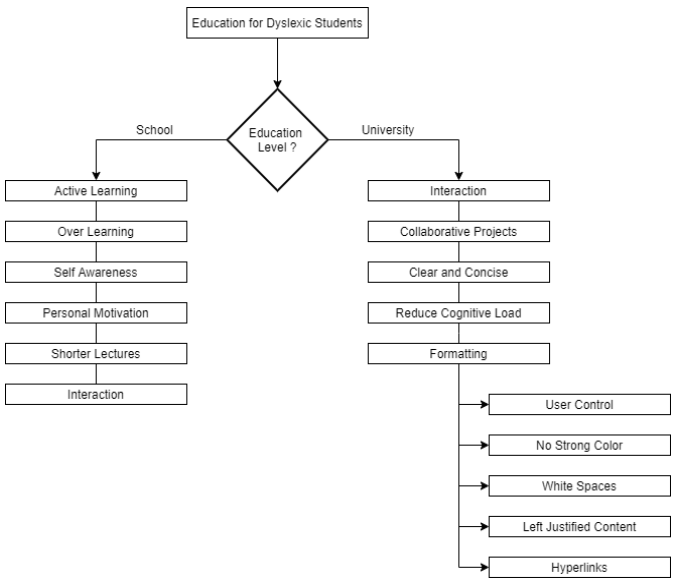

Fig -3: Proposed Teaching Methodology

The above figure 3 portrays the division of the Education Level into two phases, School level education and University level education. In school-level education, the overall development of the dyslexic students, as well as normal students, is considered. The overall growth in the school level can be ensured with some methodologies such as active learning, overlearning, self-awareness, personal motivation, shorter lecture span, and student to student interaction. University-level education focuses more on

programming classes. Some measures that may enhance the understanding of a dyslexic student in a programming class are interacting with fellow students, collaborative projects, clear and concise content of notes and presentations, reduction in cognitive load using proper formatting and layout of the presentation.

3.1 School Level Proposed Teaching Methodology

School-based education is more concerned with the overall development of the student. Normally, at this early age, dyslexic students are unable to remember the order of the days in a week [36], they are more prone to mistakes than the other normal students. Reading and comprehension speed is observed to be significantly low in dyslexic students. All this can cause a dyslexic student to experience a lack of self-confidence. Some of the following methods can help in bridging this lack of confidence and provide an opportunity for students to grasp and learn, more efficiently.

Active Learning: The inclusion of active learning methodologies in the classroom will aid dyslexic students in applying their right-brained skills to understand the concepts better. Active learning also helps in a better understanding of the subject in the case of normal students which results in better grades overall.

Overlearning: Overlearning has satisfactory improvement in the retention of information. Dyslexic students can overcome their weak retention abilities by applying various overlearning techniques. Schendel and Hagman (1982) also bolstered the concept of overlearning by experimenting with a group of soldiers. The soldiers were asked to do a military task which required a procedure to follow. Schendel and Hagman observed that the group which overlearned the technique produced 65% fewer errors when they were tested after 8 weeks [9].
Krueger (1930) also studied a maze experiment with 50% overlearning, 100% overlearning, or 200% overlearning. He observed that as the overlearning increased the retention level also increased [10][7].

Self-Awareness: Dyslexic students may be intimidated by the performance and learning abilities of normal students around them. Self-awareness in the learning procedures will provide an opportunity for students both dyslexic and normal to analyze their studies, understand their weaknesses, and then eventually create an independent study plan for themselves to improve on their weaker areas. Self-awareness may cultivate a new interest in students which they can pursue from thereon [7].

Personal Motivation: One of the key aspects of personal development is personal motivation. Self-awareness is followed by personal motivation. Students will be more motivated to wholeheartedly complete the work they are assigned and eventually increasing their overall grades. Teaching methodologies should adopt confidence-boosting exercises and teaching methods, active learning will play a major role in this. Due to this dyslexic student will be less anxious and more prepared [7].

Shorter Lectures: Dyslexic students have a shorter concentration ability. Usually, university lectures last up to an hour, during this time frame the dyslexic student finds it difficult to cope up with the pace of the lecture. Grasping level will increase if there are shorter time span for lectures [7].

Interaction: A positive atmosphere for interaction should be a key feature of a classroom lecture. Dyslexic students should work and interact with other normal students to share ideas and concepts. A smaller strength of students in a classroom will provide an opportunity for the lecturer to personally pay attention to each and every student according to the student's needs [7].

3.2 University Level Programming Methodologies

Programming lectures can be extremely stressful and laborious for students especially with learning disabilities like Dyslexia. The average duration of a university lecture is about 60 minutes. Dyslexic students find it difficult to concentrate for such a long period of time. A programming class requires learning simultaneously with many other tasks such as programming, drawing flowcharts [11]. This requires a lot of effort and the following measures can be taken to ease the strain of dyslexic students.

Interaction: Interaction between dyslexic programming and normal programming students should be encouraged so that ideas can be shared. Interaction with other students will boost their confidence in programming and the subject in general.

Collaborative Projects: Dyslexic students tend to commit spelling mistakes while writing simple code. In contrast, they are very good with system analysis and software engineering, they are able to see connections between different systems easily due to their strong right-brain visual skills better than normal students [12]. Thus, a collaborative project development would bring out the best in both dyslexic and normal students.

No Strong Color: Strong color and abstract background may unsettle dyslexic programmers while they concentrate on the slide. The strong background color or pattern makes it very perplexing to read [13][14]. The use of different colors should be used just to differentiate between different sections of the write-up, preferably red. When choosing colors, use a dark text on a pale background [15].

Fonts: Dyslexic students face problems with some of the fonts, the following are the suitable fonts [29]

1. Dyslexie [30]
2. Open Dyslexic [31]
3. Sylexiad [32]
4. Arial [33]
5. Comic Sans [34]
6. Verdana [35]

Left Justified: The text should be left-justified as this gives a common structure to the write-up or notes. Change in the structure of the text makes it difficult for a dyslexic to concentrate [13][15].

Clear and Concise: Use clear and concise language for lecture slides and notes [13][15].

User Control: Allow the user to adjust the screen according to his convenience. Some animations and blinking text can cause serious strain on a dyslexic student which leads to loss of concentration and reduced grasping ability [13][15]. Thus, allowing users to turn these animations can help in a better understanding of the topic as animations cause a disturbance [16].

Reduce Cognitive Load: If the cognitive load is reduced the student can concentrate better and focus his/her attention on the topic in hand. Using consistent layouts and formats may ensure a reduction in cognitive load [13][14][16].

White Space: Text should not appear cluttered on the page as it will hinder the student's ability to process information [13][15].

Hyperlinks: Keep all the hyperlinks at the last section of the notes or slide instead in-line hyperlinks as it compromises the text [13][14]. Use hover over comments which provide a description of the hyperlink, why it was used, and where it will take the user [14][15].

4. STRENGTHS OF DYSLEXIC STUDENTS

Dyslexic students are efficient in right-brained activities. This paves a way for them to excel in the following activities:

1. Some students are especially skilled at spatial reasoning and can put together a 3D spatial perspective. This can be useful in 3D game programming or model generations [17].
2. Dyslexic people can be exceedingly excellent in creative fields like music and arts [26].
3. Some dyslexics are extraordinarily good at finding connections and viewing an idea from various perspectives which is not visible to most of the normal people; this skill is called interconnected reasoning [26]. This can seem to be useful in finding problems while a project/program is in its developmental stage.
4. Some dyslexics show narrative reasoning, that is they tend to remember facts as experience or stories instead of abstractions [20].
5. Many dyslexic students struggle in schools, but they grow up to be admirable entrepreneurs [17].
6. They can find a solution to incomplete or changing facts also known as dynamic settings [20].
7. People with dyslexia have better sight than most, meaning they'll quickly absorb a full scene. Although it may be hard to focus on individual words, dyslexia seems to create it easier to work out outer edges [17].

The above-mentioned strengths of a dyslexic person can play an essential role in programming might even be revolutionary if the right tools are available for them to showcase their strengths in the field of programming.
About 80-90% of dyslexics show narrative reasoning and dynamic reasoning as strengths that can play an essential role in programming.

Making programming tools for nurturing this tremendous talent is necessary but we can also work on creating jobs that specialize in the fields which these specially wired people are good at.

5. VISUAL PROGRAMMING LANGUAGES (VPL)

5.1 Introduction

A Programming Language that uses graphical elements and figures for the development of programs is called a Virtual Programming Language (VPL) or an Executable Graphics Language. Generally, VPLs include graphical elements, symbols, icons, and text in two or more dimensions for the purpose of designing a program. VPLs are different from GUI-based programming languages as GUI-based languages only provide graphical program authoring services and have textual code. Arrangement of graphical objects lets the developer the ability to frame a set of instructions, thus minimizing the need for text-based input. A VPL can have several types, such as diagramming language, icon-based language, and form-based language [37].

VPLs focus on making programming easier for novices and for those who face challenges while reading and writing. VPLs support programmers in the following levels:

1. **Syntax:** Icons/blocks, forms, and diagrams are used to reduce or even completely eliminate the potential of syntactic errors.
2. **Semantics:** Programming primitives may be disclosed through some mechanism in a VPL.
3. **Pragmatics:** Programming primitives might be supported by VPLs to support the study of what a program means [38].
4. **Efficiency:** Reduction of text-input can help the dyslexics to design their programs without needing to interact with the text. Therefore, making their code better and more efficient.
5. **Reducing Cognitive-load:** Reduction of the text plays an important role in minimizing the cognitive-load and jumbling of letters.

Several reports from creative and intelligent people that they are unable to convert their visual ideas into text-based programs. This deprives them of good programming, due to which they could not solve many convoluted problems. Considering this fact and to reduce the learning curve text-based programming should be replaced with an artificial visual representation of visual ideas. Hence assuaging the

problem by encouraging the development and use of Visual Programming Languages [19].

Visual Programming Languages can be classified as follows:

1. Purely visual languages
2. Hybrid text and visual systems
3. Programming-by-example systems
4. Constraint-oriented systems
5. Form-based systems

1. Purely visual languages:
The programmer creates the program using visual graphics. In purely visual language the visual representation is not converted into intermediate text-based code. Debugging and execution follow in the same visual environment [19].

2. Hybrid text and visual systems:
Similar to purely visual languages, Hybrid text and visual languages support visual representation and in addition, it also converts visual representation to intermediate high-level textual code [19].

3. Programming-by-example systems:
In this method, a software-system is programmed using its own interface. An example of what to do is given to the system so as to write a program. Remembering the sequence of actions, the system is able to perform the actions repeatedly [29].

4. Constraint-oriented systems:
These can be used to create simulation models. The designer can model real-world objects with some constraints in the visual representation. The objects are designed to obey natural laws [19].

5. Form-based systems:
The ideas from spreadsheets can be used to develop programming systems. This is exactly what form-based languages do, groups of interconnected cells are altered over time and the programmer is allowed to visualize program execution [19].

The most important of them all is the purely visual language as this includes only graphical components that dyslexic programmers are comfortable using and has no words or complex spellings that are problematic. The program in such languages is compiled directly from its visual representation and is never converted to any textual intermediate form. This is why it's beneficial for the dyslexic programmers as even errors can be solved without having to use text.

Some theoretical advancements in the field of Visual Programming Languages produced some definitions as follows:

1. **Icon:** An object in a VPL with the dual representation of a logical part (backend) and the physical part (an image)
2. **Iconic sentence:** It is an arrangement of icons in the visual system
3. **Visual language:** A set of such iconic sentences written using a given syntax or semantics
4. **Syntactic analysis**(parsing)**:** An analysis of the iconic sentence to parse the underlying structure
5. **Semantic analysis**(interpretation)**:** To understand the meaning behind the icons. [19]

5.2 Tools that can simplify programming activities

Advancements and research in the field of Computer Science and an increase in processing power have led to the addition of graphics for development purposes. Development in Visual Programming Languages has burgeoned for a few decades. This resulted in the production of several tools for Visual Programming. Some of these tools have been mentioned in Table 1.
The following table presents some VPLs, their applicability and purposes in development.

Table -1: Visual Programming Languages

Sr. No	VPLs	Applicability	Purpose
1.	Fabrik	No	General
2.	Scratch	Yes	Special
3.	Flip	Yes	Games
4.	Alice	Yes	Animation
5.	App Inventor	Yes	Mobile App

* Applicability in the table stands for the practical use of the language and it has two values either Yes or no.

Some of the tools used for Visual Programming have been discussed below.

1. Fabrik

While several tools have been developed targeting visual programming environments, a kit called "Fabrik" was developed as early as the mid-1980s by Smalltalk at the Advanced Technology Group of Apple. This kit wires together, user-interface components and computation to build new components and useful applications.

Fabrik used a bidirectional data-flow connection approach as a shorthand for multiple paths of flow. Arbitrary objects could be computed by Fabrik as being built on object-oriented foundations. Fabrik uses several libraries that provide functionalities like arithmetic, string, file access, logical manipulations, and user-interface generation [21].

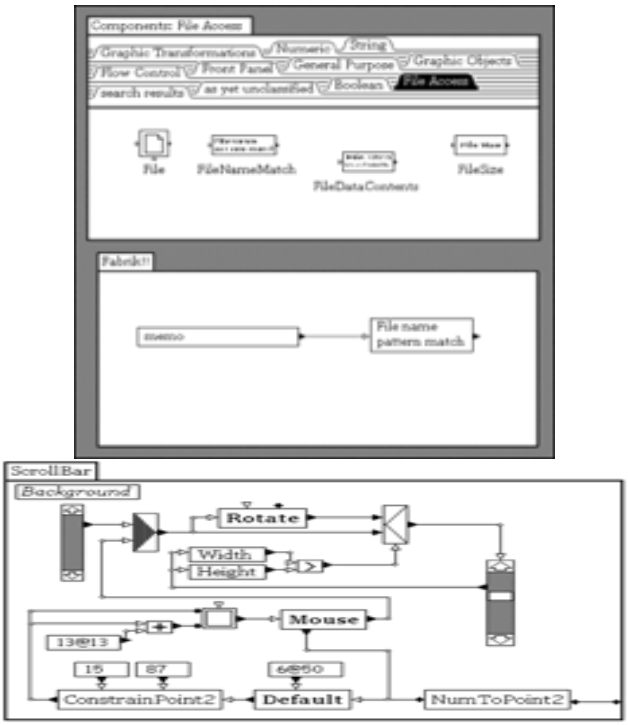

Fig -4: The Fabrik interface

2. Scratch

The approach for developing scratch and its introduction to the programming world was to appeal to the people who hadn't previously imagined themselves as programmers [23].

1. Scratch has been called "the YouTube of interactive media." [23] Each day, Scratchers from around the world upload more than 1,500 new projects to the site, with source code freely available for sharing and remixing.
2. With its easy computation and graphical interface, scratch is an ideal language for dyslexic people to get into programming.

Fig -5: Sample Scratch scripts

The scripts are available as graphical blocks which can be processed with minimal textual use.

Scripts can be easily created by dragging the graphical blogs in the script area which makes it convenient for any user working with scratch.

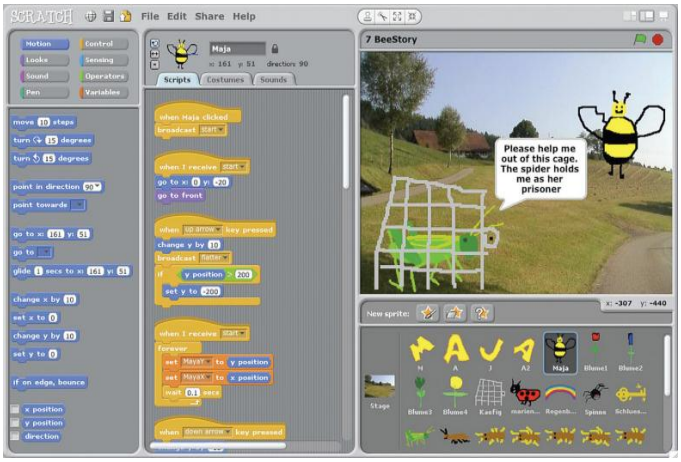

Fig -6: Scratch User Interface

Figure 6 shows a complete graphical user interface of scratch suited for dyslexic programmers.

3. Flip: a bi-modal programming language for game creation

Flip is aimed at inspiring 11-15-year-old students to develop 3D role-playing games [25]. Thus, cultivating their interest in programming from an early age.
The two parts of the flip include:

1. A visual language (a block-based graphical language similar to other VPLs)
2. A dynamically updating textual language version of the flip script under creation by the user.

This natural language/programming language pairing is a unique feature provided by flip which helps the user to use their familiarity with natural language for solving errors.

Flip aims to help young people develop computational skills by allowing them to write scripts for creating narrative-based computer games. A combination of two factors makes flip unique as compared to other VPLs and helps students develop computing skills [26].

Flip provides a totally graphical block-based user interface which is beneficial for dyslexic programmers.

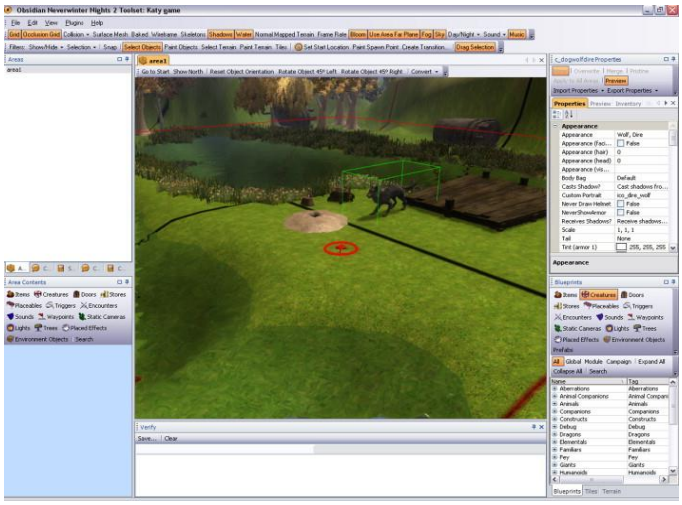

Fig -7: Flip Electron Toolset

Figure 7 shows RPG game environment developed using Flip Electron toolset. This toolset provides user friendly environment for game development similar to that of Unity and Unreal Development Kit (Epic games) which are well built and focused on high-end game development. Generally, developers using flip electron toolset don't need to program their character or game environments exclusively instead they can choose appropriate designs made available by the toolset itself [25].

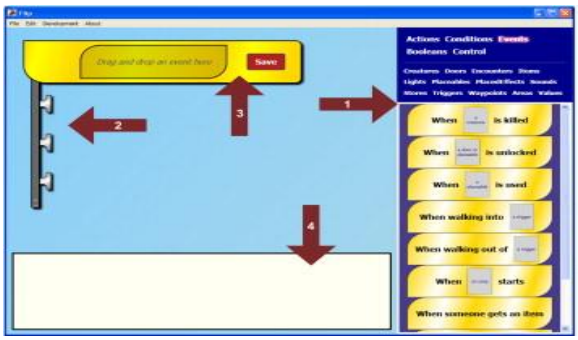

Fig -8: Flip scripts

In figure 8,

1. The Block Box contains the blocks used to create scripts. Blocks of the currently selected category are displayed in the lower panel of the column.
2. The Spine, where the script is composed. The silver pegs are used to attach blocks to the spine. The blocks attached to the spine dictate what the script will do.
3. The Event Slot, where a single event block is placed to dictate when the script will execute.
4. The Natural Language (or 'plain English') box. As the script is built up, a natural language equivalent of the script automatically appears here.

6. CONCLUSION AND FUTURE SCOPE

According to the survey conducted above, we conclude that changing teaching methodology can engender a new way of learning for dyslexic students. These changes will not only be beneficial for dyslexic students but also normal students. We also discussed how VPL's can improve a dyslexic student's skills and thus in turn creating an interest in software development. A staggering 17% of the population is dyslexic and thus using the above methodologies they can aim for better opportunities overall. Considering the research and tools available for VPLs, it can be assumed that programming can be made easy for dyslexics by using visual methods. The use of VPLs in the development of programs can give a rise in the number of dyslexic programmers. In the future, we aim to research how VPL can be integrated with text-based languages so that dyslexic programmers who prefer VPL can work in cohesion with industry programmers who work in traditional text-based programming languages.

7. REFERENCES

[1] Dyslexiahelp.umich.edu. n.d. Frequently Asked Questions. <http://dyslexiahelp.umich.edu/answers/faq>.

[2] Ugc.futurelearn.com. n.d. <https://ugc.futurelearn.com/uploads/files/79/1c/791c7fe2-9b8d-496c-b17c-21b48bce1cbb/kormosandsmithchapter2.pdf>.

[3] International Dyslexia Association. n.d. Definition of Dyslexia - International Dyslexia Association. <https://dyslexiaida.org/definition-of-dyslexia/.>.

[4] Frith, U. (1999), Paradoxes in the definition of dyslexia. Dyslexia, 5(4), pp.192-214.

[5] Mcleod, S. (2010), Long-Term Memory | Simply Psychology. <https://www.simplypsychology.org/long-term-memory.html>.

[6] Mcleod, S. (2012), Working Memory Model | Simply Psychology. <https://www.simplypsychology.org/working%20memory.html>.

[7] Stienen-Durand, S. and George, J. (2014), Supporting Dyslexia in the Programming Classroom. Procedia Computer Science, 27, pp.419-430.

[8] Brain Made Simple (2019), Hemispheres | Left & Right Hemispheres Roles, Facts & Information. <https://brainmadesimple.com/left-and-right-hemispheres>.

[9] Schendel, J. and Hagman, J. (1982), On sustaining procedural skills over a prolonged retention interval. Journal of Applied Psychology, 67(5), pp.605-610.

[10] Driskell, J., Willis, R. and Copper, C. (1992), Effect of overlearning on retention. Journal of Applied Psychology, 77(5), pp.615-622.

[11] Libguides.reading.ac.uk. n.d. Libguides: Studying with Dyslexia and Other Specific Learning Difficulties: Tips for Lectures and Seminars. <https://libguides.reading.ac.uk/dyslexia/lectures>.

[12] Stein, J. (2018), <https://www.bcs.org/content-hub/why-dyslexics-make-good-coders>.

[13] Powell, N., Moore, D., Gray, J., Finlay, J. and Reaney, J. (2004), Dyslexia and learning computer programming. ACM SIGCSE Bulletin, 36(3), p.242.

[14] Imsglobal.org. n.d. Accessibility | IMS Global Learning Consortium. <https://www.imsglobal.org/activity/accessibility>.
[15] Rainger, P. (2003), <https://users.dimi.uniud.it/~giorgio.brajnik/dida/psw/galleria/05/vallega/relaz-1/03-04-01_dyslexic.pdf>.
[16] Fuertes, J., Gonzalez, L. and Martinez, L. (2016), Characterization of Programmers with Dyslexia. <https://www.researchgate.net/publication/304822013_Characterization_of_Programmers_with_Dyslexia>.
[17] Nessy US. n.d. 9 Strengths of Dyslexia - Nessy US. <https://www.nessy.com/us/parents/dyslexia-information/9-strengths-dyslexia>.
[18] New England Dyslexia Solutions. n.d. Adultdyslexiacharacteristics. <https://www.ne-dyslexia.com/adult-dyslexia-characteristics.html>.
[19] Boshernitsan, M. and Downes, M. (2004), Visual Programming Languages: A Survey. <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2004/CSD-04-1368.pdf>.
[20] WIRED (2011), Q&A: The Unappreciated Benefits of Dyslexia. <https://www.wired.com/2011/09/dyslexic-advantage>.
[21] Ingalls, D., Wallace, S., Chow, Y., Ludolph, F. and Doyle, K. (1988), Fabrik: a visual programming environment. ACM SIGPLAN Notices, 23(11), pp.176-190.
[22] Field, L. and Kaplan, B. (1998), Absence of Linkage of Phonological Coding Dyslexia to Chromosome 6p23-p21.3 in a Large Family Data Set. The American Journal of Human Genetics, 63(5), pp.1448-1456.
[23] Resnick, M., Maloney, J., Hernández, A. and Rusk, N. (2009), <https://web.media.mit.edu/~mres/papers/Scratch-CACM-final.pdf>.
[24] Appinventor.mit.edu. n.d. MIT App Inventor | Explore MIT App Inventor. <https://appinventor.mit.edu>.
[25] Howland, K. and Good, J. (2015), Learning to communicate computationally with Flip: A bi-modal programming language for game creation. Computers & Education, 80, pp.224-240.
[26] Touch-type Read and Spell (TTRS). n.d. 3 Dyslexia Strengths You Should Know About. <https://www.readandspell.com/dyslexia-strengths>.
[27] Keung, L. (2019), Best Fonts for Dyslexia. <https://design.tutsplus.com/articles/best-fonts-for-dyslexia--cms-33358>.
[28] Alice.org. n.d. Alice – Tell Stories. Build Games. Learn to Program. <https://www.alice.org>.
[29] ResearchGate (1984), Programming by Example. <https://www.researchgate.net/publication/35761457_Programming_by_Example>.
[30] Dyslexiefont.com. n.d. Dyslexia: It's All Greek To Me. <https://www.dyslexiefont.com/en/blog/30/dyslexia-its-all-greek-to-me>.
[31] Opendyslexic.org. n.d. Opendyslexic. <https://opendyslexic.org/>.
[32] Snyder, J. (2020), Technology Focused <http://www.robsfonts.com>.
[33] Fonts.com. n.d. Arial® Font Family | Fonts.Com. <https://www.fonts.com/font/monotype/arial>.
[34] Fonts.com. n.d. Comic Sans® Font Family | Fonts.Com. <https://www.fonts.com/font/microsoft-corporation/comic-sans>.
[35] Fonts.com. n.d. Verdana® Font Family | Fonts.Com. <https://www.fonts.com/font/microsoft-corporation/verdana>.
[36] TheSchoolRun. n.d. How Does Dyslexia Impact on A Child's Learning? <https://www.theschoolrun.com/how-does-dyslexia-impact-childs-learning>.
[37] Feijs, L., 1990. The interface description language: definition and use. Science of Computer Programming, 14(1), pp.106-109.
[38] En.wikipedia.org. n.d. Visual Programming Language. <https://en.wikipedia.org/wiki/Visual_programming_language>.