

---

## Model Gaya Belajar Perceptual Yang Sesuai Untuk Pemahaman Programming

Stephanie Bj Parera<sup>1</sup>, Stenly Richard Pungus<sup>2</sup>

<sup>1,2</sup>Universitas Klabat; Jl. Arnold Mononutu, Sulawesi Utara, Indonesia 95371

Telp: 0431 891035 Fax: 0431 891036

e-mail: www.unklab.ac.id

<sup>3</sup>Program Studi Teknik Informatika, Fakultas Ilmu Komputer UNKLAB, Manado

e-mail: \*<sup>1</sup>11310404@student.unklab.ac.id, <sup>2</sup>stenly.pungus@unklab.ac.id

### Abstrak

Paper ini melakukan kajian tentang model gaya belajar perceptual, yang digunakan untuk membantu pelajar dalam memahami programming. Gaya belajar adalah bagaimana manusia mengolah informasi yang diperoleh menjadi pengetahuan dengan cara tertentu. Peneliti menemukan adanya kesulitan dalam memahami konsep dasar programming bagi para pemula. Gaya belajar yang diterapkan belum sesuai untuk memahami konsep programming yang dinamis. Gaya belajar yang diteliti dalam kajian ini adalah gaya belajar jenis perceptual. Ada 3 gaya belajar perceptual yang umum dilakukan yaitu visual, auditorial, dan kinestetik (VAK). Setiap gaya belajar mempunyai hubungan dengan aktivitas yang biasa dilakukan manusia, mulai dari melihat, mendengar, serta bergerak. Pemilihan gaya belajar diperlukan dalam mendukung terciptanya proses belajar mengajar yang efektif, begitu juga dalam pembelajaran programming. Oleh karena itu, peneliti bertujuan untuk menjelaskan karakteristik dari setiap gaya belajar dan tanggapan dari para ahli tentang belajar programming. Dalam kajian ini, peneliti menggunakan metode studi literatur dalam memperoleh informasi. Hasil dari kajian ini adalah model gaya belajar yang mencakup gaya visual, auditorial dan kinestetik dalam proses pembelajaran.

**Kata kunci**—gaya belajar, perceptual, VAK, programming, pemahaman programming

### Abstract

*This paper making a study about perceptual learning styles, which is used to help the student for understanding the programming. The learning style is about how human being process the information that they had to something learnable. Researcher found that there is such an obstacles in understanding the basic concept of programming for student. The learning style which is applying is not appropriate to understand the dynamic concept of programming. Perceptual learning style is being observed in this study. There are three kinds of perceptual learning style that similar such as visual, auditorial, and kinesthetic (VAK). Every learning style has connection with our daily activities that human being could do like seeing, hearing, along with moving. The selection of learning style are supporting to create the effectiveness of teaching and learning process, and so do the programming. That is why, researcher aiming to explain the characteristics of every learning style and what does the other researcher note about it. In this research, researcher using literature review method to get the information. The conclusion of this study is a model of learning style that involving visual, auditorial, and kinesthetic in learning process.*

**Keywords**—learning style, perceptual, VAK, programming, understanding of programming

### 1. Pendahuluan

Belajar merupakan aktivitas mental yang melibatkan kemampuan otak dalam menyerap, mengolah dan menyampaikan informasi [1]. Gaya belajar adalah karakteristik dari proses belajar yang dilakukan orang pada umumnya. Setiap orang memiliki gaya belajar yang berbeda-beda. Pada penelitian [2] model gaya belajar dibagi dalam 5 kategori, yaitu : 1) *Environmental*; 2) *Emotional*; 3) *Sociological*; 4) *Physiological*; 5) *Psychological*. Perceptual adalah salah satu contoh elemen pada kategori physiological. Perceptual adalah model gaya belajar yang dapat meningkatkan kemampuan pelajar untuk memberi respon terhadap lingkungan sekitarnya [3]. Contoh manfaat gaya belajar *perceptual* adalah proses untuk mengumpulkan informasi disesuaikan dengan organisme yang menggunakan informasi tersebut. Kehadiran gaya belajar ini membantu anak-anak untuk belajar lebih mudah, dengan kesulitan

---

yang dihadapi, atau tidak sama sekali. Semua bergantung pada bagaimana anak-anak tersebut diperkenalkan pada informasi yang baru atau menantang. Gaya belajar *perceptual* mencakup 3 gaya belajar yang umum digunakan oleh para pelajar, yaitu Visual, Auditorial, dan Kinestetik. Pelajar dengan gaya belajar visual, belajar dengan melihat bentuk-bentuk atau gambar. Hal ini berbeda dengan pelajar auditorial yang mengandalkan kemampuan untuk mendengar. Pelajar dengan gaya belajar kinestetik lebih suka belajar dengan cara terlibat langsung [1]. Gaya belajar ini dapat diterapkan dalam pembelajaran *programming*.

Menurut McCracken [4], *programming* pada dasarnya adalah proses menerjemahkan dari bahasa yang menyenangkan bagi manusia ke bahasa yang menyenangkan bagi komputer. Belajar bahasa *programming* memiliki manfaat, contohnya di bidang bisnis. Besarnya permintaan pasar terhadap kehadiran aplikasi membuat para *developer* berlomba untuk membuat aplikasi. Siswa atau mahasiswa yang mempelajari *programming* akan belajar proses pemecahan masalah dengan menggunakan konsep *programming*. *Programming* sering dianggap sulit untuk dipahami. Pelajaran *programming* di universitas biasanya memiliki tingkat *drop-out* tertinggi dan sering dianggap sebagai pelajaran paling sulit [5]. Para mahasiswa khususnya yang belum berpengalaman, mempunyai masalah dalam membaca, menangkap, menulis dan merancang satu bagian kode yang sederhana [4]. Hal ini mengakibatkan para mahasiswa menjadi malas dalam mengerjakan tugas dan tidak memiliki keberanian serta niat dalam belajar. Kurangnya pemahaman terhadap dasar *programming* membuat para mahasiswa hanya berharap simpati dari dosen atau guru.

Julukan bagi orang yang berprofesi sebagai pembuat program adalah *programmer*. *Website* [entrepreneur.com](http://entrepreneur.com) [6] membuat sebuah ke artikel yang menjelaskan tentang beberapa karakteristik dari *programmer* yang baik, di antaranya adalah kemampuan teknikal yang hebat, kesediaan untuk terus belajar, mentalitas dalam kerja kelompok, mempunyai kemampuan menyelesaikan masalah, kemampuan untuk *debugging*, dan mempunyai minat yang mendalam pada pekerjaan yang sedang dijalani. *Programmer* yang baik berkaitan dengan peranan bahasa pemrograman. Menurut [website.ustudy.in](http://website.ustudy.in) [7] terdapat lebih dari 3 karakteristik bahasa pemrograman. Contohnya: *readability*, *generality*, *Cost*, *Efficiency*, *familiar notation* dan *error-checking*. Berdasarkan karakteristik yang ada, peneliti melihat adanya hubungan antara standar seorang *programmer* dikatakan baik dan bahasa pemrograman yang digunakan dalam proses pengembangan program. Peneliti berfokus pada satu karakteristik dari *programmer* yaitu kemampuan untuk *debugging* dan satu karakteristik bahasa pemrograman yaitu *error-checking*. Kemampuan untuk *debugging* atau *debugging skills* dibutuhkan saat *software* yang dibuat tidak berjalan sesuai dengan harapan, *programmer* diharapkan langsung ke inti masalah dengan cepat dan efektif. *Programmer* akan melakukan investigasi pada kode dan mengkaji masalah yang mungkin muncul sampai solusi ditemukan. Karakteristik *error-checking* pada bahasa pemrograman dapat mempermudah *programmer* dalam mencari tahu masalah yang terjadi dalam kode. Banyak bahasa pemrograman tingkat tinggi memberlakukan *error-checking* pada saat *compile* dan *run* atau eksekusi program.

Dalam rangka mendukung penelitian ini, maka penulis menggunakan metode studi literatur dalam memperoleh informasi. Studi literatur dilakukan dengan menganalisis beberapa jurnal atau penelitian yang berkaitan dengan masalah yang diangkat penulis. Untuk memperoleh informasi yang baik maka peneliti menggunakan cara *summarize*, *compare* dan *contrast* terhadap penelitian-penelitian yang dijadikan sumber dalam penelitian ini. *Summarize* merupakan cara pengumpulan data dengan menuliskan kembali suatu literatur menggunakan kalimat peneliti. *Compare* adalah cara melakukan tinjauan literatur dengan mencari kesamaan antara literatur, sedangkan cara *contrast* adalah dengan mencari perbedaan antara literatur atau sumber yang lain.

Gaya belajar tertentu bisa membantu pelajar dalam mempelajari *programming*. Oleh karena itu, peneliti ingin melakukan analisa terhadap gaya belajar yang sesuai untuk belajar *programming*. Gaya belajar yang dianalisa dalam *paper* ini adalah gaya belajar *perceptual* yang meliputi Visual, Auditorial, Kinestetik. *Programmer* yang sudah berpengalaman dan jurnal-jurnal *online* yang berkaitan dengan penelitian ini dijadikan sebagai sumber untuk penelitian ini.

## 1.1 Jenis Gaya belajar Perceptual

### a. Visual

Gaya belajar ini mengandalkan gambar atau objek visual lain dalam memperoleh informasi.

Orang yang mempunyai gaya belajar visual memiliki ciri sebagai berikut [1] :

1. Teratur, memperhatikan segala sesuatu, menjaga penampilan
2. Mengingat gambar, lebih suka membaca daripada dibacakan
3. Membutuhkan gambaran dan tujuan tertentu.

Berdasarkan ciri yang ada maka beberapa karakteristik pembelajaran yang sesuai adalah :

1. Dosen berdiri tenang saat menyajikan segmen informasi, dan bergerak perlahan di antara segmen tersebut
2. Memberi dorongan kepada mahasiswa untuk menggambarkan informasi, dengan membuat diagram, simbol dan gambar berwarna dalam catatan mahasiswa yang menggunakan gaya belajar visual
3. Adanya tabel dan grafik yang bisa memperdalam pemahaman mahasiswa
4. Pembuatan peta pikiran atau peta konsep dalam memberikan gambaran keseluruhan suatu konsep
5. Menggunakan bahasa simbol visual dalam presentasi yang mewakili konsep utama
6. Menggunakan media pembelajaran berupa buku, majalah, poster, komputer/LCD, *flow chart*, dan tulisan dengan warna menarik.

#### b. Auditorial

Gaya belajar yang mengandalkan kemampuan untuk mendengar. Mahasiswa mengambil ide pokok dari mendengar nada, intonasi dan kecepatan suara. Orang yang mempunyai gaya belajar auditorial memiliki ciri sebagai berikut [1] :

1. Perhatiannya mudah terpecah
2. Berbicara dengan pola berirama
3. Belajar dengan cara mendengarkan, mengerakkan bibir atau bersuara saat membaca
4. Berdialog secara internal dan eksternal

Berdasarkan ciri yang ada maka beberapa karakteristik pembelajaran yang sesuai adalah :

1. memberikan informasi secara berulang-ulang
2. menggunakan variasi lokal dalam presentasi
3. menyanyikan konsep utama atau membuat lagu yang terkait dengan konsep utama
4. memberi dorongan pada mahasiswa untuk membuat konsep yang mudah untuk diingat
5. menggunakan teknik tanya jawab
6. melibatkan musik dalam pembelajaran

#### c. Kinestetik

Mahasiswa yang belajar dengan gaya belajar ini lebih suka belajar dengan cara terlibat langsung atau dengan tindakan. Mahasiswa cenderung melakukan kontak fisik dengan objek atau konsep yang dipelajari. Orang yang mempunyai gaya belajar kinestetik memiliki ciri sebagai berikut [1] :

1. Sering menyentuh orang, berdiri berdekatan, dan bergerak saat berinteraksi dengan orang lain
2. Belajar dengan melakukan sebuah tindakan
3. Menunjuk tulisan saat membaca
4. Mengingat sambil berjalan dan melihat

Berdasarkan ciri yang ada maka beberapa karakteristik pembelajaran yang sesuai adalah :

1. Memberikan tugas berupa proyek terapan
2. Menggunakan media pembelajaran atau alat bantu untuk menimbulkan rasa ingin tahu dan menekankan konsep utama
3. Memperagakan konsep sambil memberikan kesempatan kepada mahasiswa untuk mempelajarinya langkah demi langkah
4. Menciptakan simulasi konsep
5. Membuat peta pikiran atau peta konsep dengan melibatkan aktivitas fisik

### 1.2 Definisi *programming*

*Programming* adalah proses menggambarkan jadwal dari urutan operasi individual untuk menghasilkan kalkulasi [8]. Pernyataan yang sama dikemukakan oleh penelitian [5] di mana proses yang dimaksud adalah proses menerjemahkan dari bahasa yang dimengerti oleh manusia ke bahasa yang dimengerti oleh komputer. Data yang diproses berasal dari urutan dari perintah-perintah tertentu dalam komputer. Hal ini didukung dengan pernyataan oleh Wrubel [10] bahwa urutan dari perintah-perintah dasar disebut dengan program dan proses persiapan disebut dengan *programming*.

Proses persiapan berkaitan dengan urutan perintah yang akan dieksekusi secara otomatis tanpa campur tangan dari pengguna [11]. Menurut bagian pendahuluan pada buku “*Psychology of Programming*”, *programming* telah berubah dari “menjelaskan kalkulasi” menjadi “mendefinisikan fungsi-fungsi” dan kemudian “mendefinisikan dan memperlakukan objek” [12].

### 1.3 Masalah dalam belajar *programming*

*Programming* sering dianggap sulit untuk dipahami. Pelajaran *programming* di universitas biasanya memiliki tingkat *drop-out* tertinggi dan sering dianggap sebagai pelajaran paling sulit [9]. Para mahasiswa khususnya yang belum berpengalaman, mempunyai masalah dalam membaca, menangkap, menulis dan merancang satu bagian kode yang sederhana [10]. Hal ini mengakibatkan para mahasiswa menjadi malas dalam mengerjakan tugas dan tidak memiliki keberanian serta niat dalam belajar. Kurangnya pemahaman terhadap dasar *programming* membuat para mahasiswa hanya berharap simpati dari dosen atau guru.

Tabel I – Kesulitan dalam belajar *programming* [9]

Jenis-jenis kesulitan	Rata-rata
Memahami konsep dasar dari struktur <i>programming</i>	3.40
Merancang program untuk menyelesaikan tugas	3.36
Mempelajari sintaks bahasa pemrograman	3.35
Mendapat akses ke sebuah komputer	3.33
Menggunakan lingkungan pengembangan program	3.22
Mencari <i>bugs</i> dari program sendiri	2.19

Berdasarkan data pada tabel I maka didapati ada 3 kesulitan dalam *programming* yang saling berkaitan yaitu: 1) Kesulitan dalam memahami konsep dasar dari struktur *programming*; 2) Merancang sebuah program untuk menyelesaikan tugas tertentu; 3) belajar sintaks bahasa *programming*. Kesulitan atau masalah tersebut sering dihadapi oleh kaum awam yang tidak mementingkan konsep dasar dari *programming* dan langsung menggunakan konsep abstrak *programming* [13]. Hal ini disebabkan oleh kemampuan dalam *programming* mencakup lebih dari satu proses yang berbeda pada tahap awal, *programmer* harus menerjemahkannya ke dalam *algoritma* yang kemudian diterjemahkan ke dalam kode program [14].

### 1.4 Perbandingan gaya belajar *programming*

Ditinjau dari 3 gaya belajar yang telah dijabarkan sebelumnya, para ahli dan peneliti mempunyai pendapat yang berbeda-beda tentang mempelajari *programming*. Berikut adalah penelitian-penelitian yang mendukung peneliti dalam melakukan analisa [4,11,12,13,15] :

Codeacademy [15] adalah sebuah perusahaan yang bergerak di bidang pendidikan dan merupakan founder website codeacademy.com. Website ini menjadi media pembelajaran online tentang *programming* bagi semua kalangan usia. Code academy terdiri dari orang-orang yang berpengalaman di bidangnya, seperti programmer, engineer, designer, evangelist, dan analyst. Masing-masing ahli mempunyai ceritanya sendiri bagaimana mereka bisa mempelajari *programming*. Kari Tarr adalah seorang *software developer* di Airbnb yang mengawali karirnya melalui codeacademy. Kari merekomendasikan beberapa sumber yang bisa digunakan untuk belajar *programming* yaitu, Programming Ruby, Rosetta Cose, Cracking the Coding Interview dan Guide to Ruby. Sumber-sumber ini ada yang berupa buku dan juga website. Hal serupa disarankan oleh Dilys Sun. Dilys merekomendasikan buku yang berjudul Why’s Poignant Guide to Ruby dan forum tanya jawab yang tersedia di codeacademy. Selain melalui buku dan website, pakar codeacademy lainnya seperti Michael Perry, Nick Elligott dan William Ha menyarankan untuk mengadakan pertemuan dengan orang-orang yang lebih berpengalaman. Dalam pertemuan tersebut, ide-ide dasar yang

dimiliki dapat dikembangkan menjadi sebuah *project* kecil. Project kecil itulah yang menjadi dasar kesuksesan para ahli atau *programmer*. Rekomendasi dari para ahli di *codeacademy* mencakup 3 gaya belajar visual, auditorial dan kinestetik. Gaya belajar visual dapat diterapkan dengan membaca buku, membaca materi di *website* dan menonton video tutorial. Gaya belajar auditorial dapat diterapkan ketika mendengar tanggapan atau saran dari orang yang lebih berpengalaman. Materi dan saran yang didapatkan kemudian diterapkan dalam praktek, hal inilah yang menandakan gaya belajar kinestetik.

Penelitian [9] ini bertujuan untuk mencari tahu kesulitan yang dihadapi oleh para pelajar dalam mempelajari *programming*. Penelitian ini menggunakan metode survey dan melibatkan 105 pelajar *Polytechnic*. Dari hasil kuisioner, peneliti mendapati adanya situasi tertentu yang dapat membantu para pelajar.

Tabel II – Situasi yang dapat membantu untuk belajar *programming*

Situasi	Rata-rata
Dalam kegiatan praktek atau pertemuan di laboratorium	3.99
Konsultasi atau diskusi dengan dosen, senior dan teman	3.82
Dalam latihan kelompok kecil	3.74
Dalam perkuliahan	3.73
Sementara bekerja sendiri dalam tugas programming	3.39

Tabel II menjabarkan situasi-situasi yang dapat membantu para pelajar untuk belajar *programming*. Mayoritas pelajar setuju adanya praktek atau aktivitas di laboratorium dapat membantu dalam belajar *programming* dengan efektif. Belajar *programming* membutuhkan aktivitas praktikal dan latihan yang keras karena konsepnya yang dinamik [16]. Implementasi *programming* di laboratorium akan membantu pelajar untuk memahami terminologi yang sulit, dan menstimulasi minat pelajar. Bagaimanapun, aktivitas laboratorium harus diimplementasi melalui strategi yang sesuai pengalaman dan pengetahuan pelajar.

Peneliti pada penelitian [17] bertujuan untuk menjelaskan proses belajar mengajar tentang *recursive programming* yang terjadi pada pelajar maupun pengajar. Secara abstrak, rekursi dapat didefinisikan sebagai *self-referential*. Penelitian ini juga membahas tentang peran penting visualisasi dalam pengajaran *programming*. Visualisasi ini dapat diterapkan dalam bentuk animasi dan *games* yang dapat membantu orang dalam memahami *programming*. Bermain *games* edukasi, telah mencakup 3 gaya belajar sekaligus. Pengguna dapat melihat animasi dari *games* tersebut, mendengar suara animasi dan mempraktekkan informasi yang diperoleh ke dalam bentuk permainan.

Penelitian [18] ini berbentuk studi literatur dimana peneliti mengumpulkan dan mengelompokkan literatur tentang mengajar *programming* dari berbagai sumber. Secara khusus, para peneliti membagi pembahasannya ke dalam beberapa kategori, yaitu Curricula, Pedagogy, Language choice, dan Tools for teaching. Kategori *tools research* membahas tentang *tools* yang biasa digunakan dalam mempelajari *programming*. Saat ini, sebagian besar *tools* dapat diunduh dari *website*, mudah ditemukan dengan *search engines* dan didukung oleh berbagai *platform*. Dalam kaitannya dengan gaya belajar, *website* dapat dikategorikan dalam gaya belajar visual. Beberapa penelitian menunjukkan bahwa visualisasi sederhana belum cukup untuk mendukung pembelajaran

*programming*. Solusinya yaitu visualisasi ini dilanjutkan ke dalam praktek atau aktivitas yang dikategorikan dalam gaya belajar kinestetik.

Sama seperti penelitian [18], penelitian [19] juga berbentuk studi literatur dimana peneliti mendapati bahwa mayoritas *novice* dan *experts* belajar *programming* dengan praktek langsung dalam laboratorium. Frekuensi latihan atau praktek merupakan faktor penentu kesuksesan dalam belajar *programming*. *Novice* dan *experts* memiliki frekuensi praktek yang jauh berbeda. Perbedaan ini mengindikasikan pengalaman dalam belajar *programming*. Pengalaman ini bisa berupa kesuksesan dalam mengembangkan sebuah aplikasi maupun kegagalan yang terjadi selama proses belajar. Jalannya sebuah praktikum bergantung pada pemahaman terhadap teori *programming* yang didapatkan dari membaca buku atau sumber tertulis lainnya.

Penelitian [20] adalah paper yang menjelaskan tentang keterlibatan gaya belajar yang berbeda-beda terhadap kinerja pelajar dalam pengenalan *programming*. Gaya belajar yang digunakan adalah gaya belajar menurut Felder. Para peneliti menemukan adanya kegagalan dari pengenalan *programming* untuk memotivasi pelajar. Hal ini diakibatkan oleh kepasifan dan daya saing yang memaksa pelajar serta fokus pada masalah penyelesaian algoritma dibandingkan dengan pemahaman konseptual. Untuk menyelesaikan masalah tersebut, maka para peneliti mengevaluasi hubungan antara gaya belajar tertentu dan pengaruhnya terhadap ujian dan aspek praktikal dari pengenalan *programming*. Hasil evaluasi dari penelitian ini, kemudian digunakan untuk rancangan lingkungan pembelajaran dari para peneliti. Para peneliti melakukan sebuah percobaan untuk menguji hipotesa-hipotesa bahwa jenis gaya belajar yang berbeda-beda mempunyai dampak yang berbeda-beda pula terhadap pelajaran pengenalan *programming*. Percobaan ini melibatkan para pelajar yang menggunakan bahasa Java sebagai bahasa *programming* dalam kelas pengenalan *programming*. Pelajar dibagi ke dalam dua golongan, keduanya mempunyai ujian dan tugas sama yang diajarkan oleh instruktur yang sama pula. Hasil dari percobaan ini dapat digunakan dalam dua cara. Cara yang pertama yaitu melanjutkan pelajaran dengan cara mengajar yang sama seperti sebelumnya, dan memberikan uji gaya belajar terhadap pelajar yang baru masuk lalu menempatkannya ke dalam program dari peneliti berdasarkan hasil uji. Cara ini bertentangan dengan maksud dari tujuan pendidikan universitas yang adalah memperluas kesempatan bukan membatasinya. Cara yang kedua yaitu membuat material atau bahan yang bisa menyampaikan informasi dan kemampuan dari pelajaran pengenalan *programming*. Adanya material membuat pelajar dapat mencerminkan gaya belajarnya masing-masing dan memperkuat model pengajaran dari pengajar maupun pelajar.

Penelitian [20] dan penelitian [21] sama-sama meneliti tentang relasi antara gaya belajar dengan pelajar khususnya di bidang *programming*. Penelitian [21] menggunakan gaya belajar menurut Kolb. Gaya belajar menurut Kolb adalah 1). *The accomodating style*: Merasakan dan melakukan; 2). *The diverging style* : merasakan dan melihat; 3). *The assimilating style* : berpikir dan melihat; 4). *The converging style* : Berpikir dan melakukan.

Tabel III–Kelompok-kelompok KLSI dari pelajar yang terdaftar [22] cited in [21]

Mata kuliah	Jumlah pelajar	<i>Accommodator</i>	<i>Diverger</i>	<i>Assimilator</i>	<i>Converger</i>
<i>Art &amp; Commerce</i>	198	13%	13%	47%	27%
<i>Programming</i>	48	4%	8%	42%	46%

Tabel III menunjukkan bahwa mayoritas pelajar yang mengambil mata kuliah *programming* menggunakan cara belajar *converger*. *The converging style* adalah gaya belajar yang sesuai untuk

menyelesaikan masalah-masalah praktikal, dan biasanya menggunakan simulasi dalam proses pembelajaran. Pelajar yang menggunakan gaya belajar ini adalah pelajar yang memahami pelajaran dengan melakukan aktivitas tertentu seperti praktek.

Tabel IV – Hasil penggolongan gaya belajar

Penelitian	visual	auditorial	kinesthetic
[4]	x		x
[11]	x	x	x
[12]	x		x
[13]	x		x
[15]	x	x	x
[20]	x	x	x
[21]	x		x

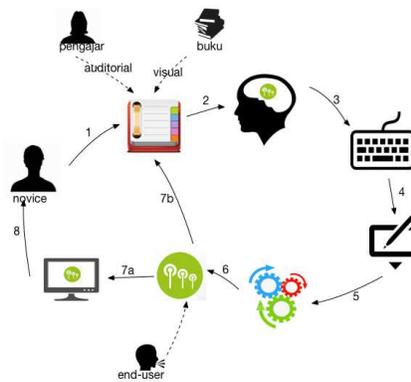
Tabel IV menunjukkan penggolongan gaya belajar berdasarkan studi yang dilakukan peneliti. Peneliti mengambil 7 sumber utama dalam menentukan gaya belajar yang sesuai untuk pemahaman *programming*. Berdasarkan tabel hasil penggolongan, gaya belajar visual dan kinestetik adalah gaya belajar yang paling banyak direkomendasikan. Proses pembelajaran yang lebih efektif dapat didapatkan dari menggabungkan 3 gaya belajar yaitu visual, auditorial dan kinestetik.

2. Metode Penelitian

Metode penelitian yang digunakan dalam kajian ini adalah studi literatur. Langkah-langkah penelitian yang dilakukan dalam kajian ini adalah pengumpulan data, pengolahan data dan penyusunan laporan penelitan. Pada tahap pengumpulan data, peneliti mengumpulkan data dari beberapa literature yang saling berkaitan dan sesuai dengan masalah yang diangkat peneliti dalam kajian ini. Setelah semua data telah diperoleh, data diolah melalui analisis dan pengujian. Analisa dan pengujian yang dilakukan peneliti menghasilkan tabel penggolongan gaya belajar perceptual dari tujuh penelitian atau sumber. Tahap terakhir yang dilakukan peneliti adalah penyusunan laporan penelitian. Laporan penelitian yang dihasilkan dalam penelitian ini berupa kajian literature. Kajian literatur ini menghasilkan model yang diusulkan peneliti dalam memahami *programming*.

3. Hasil dan Pembahasan

Sesuai dengan hasil yang diperoleh peneliti dari tabel penggolongan gaya belajar, gaya belajar visual dan kinestetik paling direkomendasikan untuk pemahaman *programming*. Namun hasil lain yang diperoleh adalah kombinasi antara 3 gaya belajar *perceptual* dapat membantu pemahaman *programming* menjadi lebih efektif. Oleh karena itu, peneliti membuat sebuah representatif model gaya belajar yang digunakan untuk memahami *programming* pada umumnya.



Gambar 1 – Model yang diusulkan untuk pemahaman programming

Gambar 1 adalah model gaya belajar *programming* yang coba peneliti gambarkan, untuk membantu pemahaman tentang bagaimana seorang *novice* atau orang awam memahami *programming*. Proses pemahaman *programming* terdiri dari beberapa tahap, yaitu :

1. *Novice* akan mencari tahu atau belajar tentang teori-teori dasar dari *programming*. Teori atau konsep tentang *programming* dapat diperoleh dari *textbook*, materi dalam bentuk *slide*, gambar, video, maupun penjelasan langsung dari pengajar. Penggunaan *textbook* dalam proses ini adalah contoh gaya belajar visual, sedangkan mendengarkan langsung penjelasan dari dosen adalah contoh dari gaya belajar auditorial.
2. Setelah *novice* mengetahui teori-teori apa saja yang ada dalam *programming*, *novice* akan memikirkan *output* apa yang bisa dihasilkan. Pada tahap ini *output* yang dihasilkan masih berupa rancangan awal.
3. Teori-teori yang sudah dipelajari dan rancangan awal *output* yang dihasilkan, mendorong *novice* untuk melakukan praktek. *Novice* akan mulai belajar menulis kode-kode program sederhana berupa kode untuk menampilkan kata di layar komputer, *selection*, *looping* dan menggunakan pemanggilan fungsi atau *method* dalam kode program.
4. Untuk membangun sebuah *software* atau *output* dari *programming*, maka *novice* membutuhkan *input*. *Input* yang dimaksud berupa variabel, nilai dan fungsi-fungsi yang akan digunakan.
5. Pada tahap ini, *novice* akan mulai proses *coding*. Semua teori, praktek dan input yang sudah diperoleh akan digunakan pada tahap proses ini. Proses ini mencakup proses *coding* dan perancangan *interface*.
6. Tahap proses akan menghasilkan sebuah *output* yang berbentuk *software*. *Output* tersebut harus melalui proses testing. *End-user* akan melakukan uji coba pada *output* dan memberikan masukan.
- 7a. Bila *output* sudah sesuai dengan tujuan maka *output* tersebut akan diimplementasi dan dapat digunakan dengan bantuan *hardware* seperti *PC*, *smartphone* dan *tablet*.
- 7b. Bila *output* belum sesuai dengan tujuan maka *novice* akan kembali ke tahap awal yaitu mengkaji ulang teori atau konsep dasar dalam *programming*.
8. Saat *output* berhasil diimplementasikan, pemahaman *programming* dari *novice* pun diperoleh. *Novice* sudah berubah menjadi orang yang memahami *programming* dengan baik atau bisa juga disebut dengan programmer pemula.

#### 4. Kesimpulan

Berdasarkan penjabaran sebelumnya, peneliti menarik kesimpulan bahwa gaya belajar kinestetik dapat mempengaruhi proses pembelajaran *programming* bagi kaum pelajar. Implementasi gaya belajar yang dimaksud dalam kajian ini adalah melakukan praktek untuk mendukung pemahaman pelajar. Pemilihan gaya belajar yang tepat dapat mendukung pemahaman pelajar tentang konsep dasar *programming*. Gaya belajar yang paling sesuai untuk pemahaman *programming* adalah kinestetik. Pelajar akan memperoleh pengetahuan dari kegiatan praktek yang dilakukan. Gaya belajar kinestetik dapat dikombinasikan dengan gaya belajar yang lain. Contoh kombinasi gaya belajar ini adalah membaca *textbook*, mendengarkan saran atau berkonsultasi dengan pengajar, dan melalui video-video tutorial.

#### 5. Saran

Berdasarkan hasil kajian ini, maka peneliti mendapati adanya faktor pendukung lain dalam belajar *programming*. Di antaranya yaitu faktor motivasi, *tools*, minat dan lingkungan. Hal inilah yang menjadi batasan dari kajian ini. Kajian ini tidak membahas tentang faktor-faktor pendukung lain selain gaya belajar perceptual yang mencakup visual, auditorial dan kinestetik (VAK). Kedepannya peneliti mengharapkan adanya penelitian lanjutan yang lebih terperinci tentang model gaya belajar yang lain dan faktor-faktor pendukung dalam belajar *programming*. Dengan adanya penelitian lanjutan, peneliti bertujuan membantu pelajar maupun pengajar yang ingin berbagi pengetahuan tentang *programming*. Pemahaman *programming* dapat dibagikan secara langsung maupun dengan bantuan teknologi seperti aplikasi sederhana.

**Daftar Pustaka**

- [1] Ariesta Kartika Sari, Analisis Karakteristik Gaya Belajar VAK (Visual, Auditorial, Kinestetik) Mahasiswa Pendidikan Informatika Angkatan 2014, Universitas Trunojoyo Madura, Bangkalan, Indonesia.
- [2] R. Dunn, A. Honigsfeld, L. S. Doolan, L. Bostrom, K. Russo, M. S. Schiering, B. Suh, H. Tenedero., Impact of Learning-Styles Instructional Strategies on Students' Achievement and Attitudes: Perceptions of Educators in Diverse Institution., *Clearing House*.
- [3] R. L. Goldstone, Perceptual Learning, *Annu. Rev. Psychol.* 1998, 49:585-612, Psychology Building, Indiana University, Bloomington, Indiana, 47405, 1998.
- [4] McCracken, M., Almstrum, V., Diaz, D., Guzdial, M. Hagan., D., Kolikant, Y.B., Laxer, C., Thomas, L., Utting, I. & Wilusz, T., A multi-national, multi-institutional study of assessment of programming skills of first-year CS students, *SIGCSE Bulletin*, 33(4), pp.125-180, 2001.
- [5] Yadin, A., Reducing the dropout rate in an introductory programming course., *ACM Inroads*, 2(4), 71-76, 2011.
- [6] John R., 2014, *15 Characteristic of a Good Programmer* [ONLINE]. Available : <http://www.entrepreneur.com/article/240644>
- [7] *Characteristics of a Programming Language* [ONLINE]. Available : <http://www.ustudy.in/node/10299>
- [8] Hartree, D.R., Calculating instruments and machines, *Cambridge University Press*.
- [9] SitiRosminah MD Derus, Ahmad ZamzuriMohamad Ali., DIFFICULTIES IN LEARNING PROGRAMMING: VIEWS OF STUDENTS, Faculty of Art, Computing and Creative Industry, Malaysia.
- [10] Wrubel, M.H., A primer of programming for digital computers., McGraw Hill.
- [11] Wilkes, M. V., Automatic digital computers, *Cambridge University Press*.
- [12] A. F. Blackwell., What is Programming?, *University of Cambridge Computer Laboratory*, pages 204-218.
- [13] E. Lahtinen, K. Ala-Mutka, and H.-M. Järvinen, A study of the difficulties of novice programmers, *ACM SIGCSE Bulletin*, vol. 37, no. 3, p. 14, Sep. 2005.
- [14] T. Jenkins, On the difficulty of learning to program, in *Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences*, 2002, vol. 4, pp. 53–58.
- [15] Code Academy, Codecademy Stories, [ONLINE] Available : <https://www.codecademy.com/stories>
- [16] A.Gomes dan A.J. Mendes, Learning to program difficulties and solutions, *International Conference on Engineering*, 2007.
- [17] C. Rinderknecht., A Survey on Teaching and Learning Recursive Programming., Department of Programming Languages and Compilers, Budapest, Hungary., Juli 2013.
- [18] A. Pears, Stephen Seidman, Linda Mannila, Maria Devlin, Lauri Malmi, Jens Bennedsen, Elizabeth Adams, James Paterson., A Survey of Literature on the Teaching of Introductory Programming.
- [19] Anthony Robins, Janet Rountree, Nathan Rountree., Learning and Teaching Programming: A Review and Discussion., Computer Science, University of Otago, Dunedin, New Zealand.
- [20] Lynda T., Mark R., John W., and Emma J., Learning Styles and Performance in the introductory Programming Sequence, Department of Computer Science, University of Wales, Aberystwyth, UK.
- [21] Vivian C., Michael J., The significance of learning style with respect to achievement in first year programming students, *ECU Publications Pre.2011*, Edith Cowan University Research Online.
- [22] L. Fowler, V. Campbell, D. McGill, and G.G. Roy, An Innovative Approach to teaching First Year Programming Supported by Learning Style Investigation, *14 th Annual Conference for Australasian Association for Engineering Education*, Melbourne, Australia, 2003.