

## **Pemanfaatan Platform Scratch dalam Pembelajaran Koding di Sekolah Dasar untuk mengasah kemampuan Computational Thinking pada Siswa**

**Wulandari<sup>1</sup>, Dwi Anisa Haftani<sup>2</sup>, Taufik Ridwan<sup>3</sup>, Dewi Indriati Hadi Putri<sup>4</sup>**

<sup>1</sup>Universitas Pendidikan Indonesia, Purwakarta

<sup>2</sup>Universitas Pendidikan Indonesia, Purwakarta

<sup>3</sup>Universitas Pendidikan Indonesia, Purwakarta

<sup>4</sup>Universitas Pendidikan Indonesia, Purwakarta

Pos-el: <sup>1</sup>wulandari18@upi.edu, <sup>2</sup>dwiah@upi.edu, <sup>3</sup>taufikridwan@upi.edu,  
<sup>4</sup>dewiindri@upi.edu

### **ABSTRAK**

Penelitian ini bertujuan untuk mengukur kebermanfaatan platform scratch dalam pembelajaran koding di Sekolah dasar untuk mengasah kemampuan Computational Thinking pada siswa. Perangsangan Kemampuan berpikir komputasi sejak dini pada dasarnya akan meningkatkan kecerdasan dan melatih otak anak agar terbiasa berpikir secara logis, terstruktur, dan kreatif. Metode penelitian yang digunakan adalah deskriptif kualitatif dengan metode analitis. Responden pada penelitian ini adalah 20 pengguna Scratch. Hasil penelitian menunjukkan bahwa dengan aplikasi Scratch dapat mempelajari strategi-strategi penting untuk memecahkan masalah, mendesain karya, mengomunikasikan ide-ide dengan melakukan pemrograman. Manfaat ini pun ditunjukkan oleh persepsi pengguna bahwa dari 20 responden sebagian besar responden sudah mengetahui mengenai computational thinking skills dan beranggapan bahwa computational thinking skills dapat dilatih melalui scratch, responden juga merasa Scratch cocok, mudah dan menarik digunakan untuk siswa SD. Lalu responden juga berpendapat bahwa keterampilan lain yang dapat dilatih melalui scratch, diantaranya : Ketelitian, kepekaan terhadap warna dan keterampilan berpikir kritis.

**Kata Kunci** :computational thinking skills, koding, scratch, Sekolah Dasar

Computational Thinking (CT) merupakan cara berpikir untuk memecahkan masalah, merancang sebuah sistem dan memahami perilaku manusia (Zahid, Maret 02, 2020). Di era saat ini komputer berperan penting dalam segala aspek kehidupan, sehingga tidak heran kalau computational thinking dijadikan dasar bagaimana cara berpikir dan memahami persoalan-persoalan kompleks yang terjadi. Bahkan pada tahun 2014, Inggris mulai memasukkan materi pemrograman pada kurikulum sekolah di tingkat dasar dan menengah. Hal ini dilakukan dengan tujuan untuk mengenalkan computational thinking kepada siswa sejak dini. Mereka beranggapan bahwa CT mampu meningkatkan kecerdasan

siswa, membuat siswa mampu lebih cepat untuk memahami teknologi yang ada di sekitar dan melatih otak siswa agar terbiasa berpikir secara logis, terstruktur, dan kreatif.

Sebenarnya konsep computational thinking di Indonesia sudah ada. Ditandai dengan masuknya Kompetisi Bebras yang diadakan di Indonesia setelah kunjungan Prof. Valentina Dagiene pada februari 2016 dan diundangnya perwakilan Indonesia untuk menjadi observer pada Workshop Internasional Bebras pada bulan Mei 2016. Bentuk kompetisi Bebras di Indonesia yaitu peserta diharuskan menjawab soal-soal yang terkait dengan pemecahan masalah, berpikir komputasional dan kreatifitas. Bahkan pada PERMENDIKBUD nomor 37 tahun 2018 computational thinking telah menjadi materi sendiri pada mata pelajaran informatika sebagai upaya untuk meningkatkan mutu pendidikan di Indonesia.

Penilaian Programme for International Student Assessment (PISA) menjadi salah satu yang bisa memberikan gambaran bagaimana kondisi mutu pendidikan Indonesia. Pada PISA terakhir (2018), Indonesia kembali mendapatkan hasil yang kurang menggembirakan dengan meraih rata-rata skor berturut-turut 371, 379, dan 396 dalam membaca, matematika, dan sains. Hasil ini pun mendapat reaksi dari MENDIKBUD, Nadiem Makarim. Beliau menyatakan akan menggunakan hasil PISA sebagai salah satu bahan evaluasi kualitas pendidikan di Indonesia. (Zahid, Maret 02, 2020)

PISA 2021 akan menjadi PISA pertama yang mengevaluasi kemampuan computational thinking siswa dengan mengintegrasikan ke dalam sub-pengukuran bidang matematika. Jika pemerintah ingin skor Indonesia meningkat di PISA selanjutnya, pemerintah harus waspada terhadap perubahan framework PISA tersebut dengan cara memberikan perhatian khusus pada aspek computational thinking siswa. Salah satunya dengan memberikan pengenalan mengenai koding sejak dini ke siswa melalui aplikasi scratch.

Scratch adalah bahasa pemrograman komputer untuk anak-anak yang dapat dilakukan dengan cara sederhana, yakni pengguna hanya melakukan drag-and-drop grafis. Kemampuan membuat program komputer merupakan bagian penting dari literasi pada hidup saat ini. Ketika kita belajar menulis kode menggunakan Scratch, kita mempelajari strategi-strategi penting untuk memecahkan masalah, mendesain karya, mengomunikasikan ide-ide. Sehingga hal ini dapat menunjang keberhasilan dalam menanamkan computational thinking skills pada siswa sejak dini. Scratch ini sebenarnya didesain untuk rentang usia 8-16 tahun, tapi juga dapat digunakan juga oleh segala

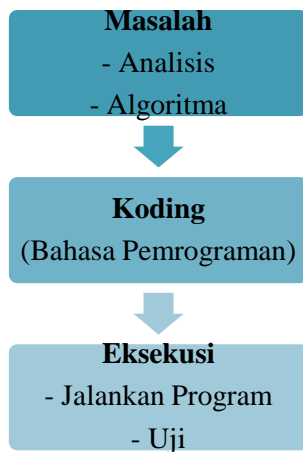
kalangan umur. Dengan Scratch, kita dapat memprogram dengan membuat cerita-cerita, permainan, animasi, dll. (<http://scratch.mit.edu>)

## **METODE PENELITIAN**

Pada penelitian ini, peneliti menggunakan pendekatan kualitatif. Pendekatan kualitatif menurut Bogdan dan Taylor merupakan prosedur penelitian yang menghasilkan data deskriptif berupa kata-kata tertulis atau lisan dari orang-orang dan perilaku yang diamati. Penelitian ini menganalisis pemanfaatan platform scratch dalam pembelajaran coding di SD untuk mempersiapkan computational thinking skills siswa. Menurut Sugiyono metode analisis deskriptif adalah statistik yang digunakan untuk menganalisis data dengan cara mendeskripsikan atau menggambarkan data yang telah terkumpul sebagaimana adanya tanpa bermaksud membuat kesimpulan yang berlaku untuk umum atau generalisasi (Sholikhah, 2016). Responden pada penelitian ini adalah pengguna scratch yang berjumlah 20 orang. Teknik pengumpulan data yang digunakan yakni studi literatur dari artikel-artikel ilmiah serta angket dalam bentuk digital (google form). Angket ini disebar melalui media sosial dan proses pengisian angket dilakukan secara mandiri.

## **HASIL DAN PEMBAHASAN**

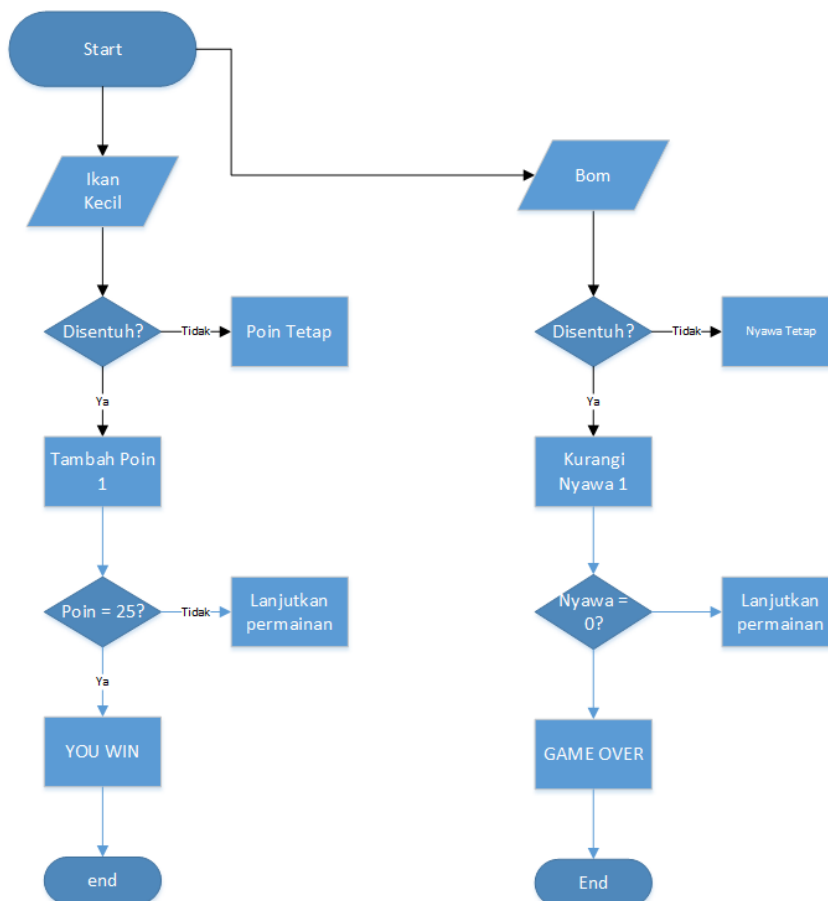
Computational thinking skills adalah kemampuan memecahkan masalah dengan cara penalaran dan analisa serta mengarahkan tindakan dengan langkah-langkah yang sistematis (Rosadi, 2020). Berpikir komputasi bukan berarti mengajarkan cara menggunakan komputer tetapi mengajarkan proses berpikir yang digunakan untuk menunjang pemecahan masalah. Untuk bisa menemukan pemecahan masalah kita bisa mengikuti cara sebuah program komputer bekerja. Dalam melakukan pemrograman terdapat program dan bahasa pemrograman. Program itu sendiri adalah kumpulan instruksi atau perintah yang disusun sedemikian rupa sehingga membentuk urutan nalar yang tepat untuk menyelesaikan suatu persoalan. Sedangkan bahasa pemrograman adalah instruksi yang memerintahkan komputer untuk menjalankan fungsi tertentu (Saragih, 2016). Berikut proses penyelesaian masalah dalam pemrograman :



**Grafik 1. proses penyelesaian masalah dalam pemrograman**

Proses pemecahan masalah ini dapat dilatih melalui aplikasi scratch. Scratch adalah bahasa pemrograman komputer untuk anak-anak yang dapat dilakukan dengan cara sederhana, yakni pengguna hanya melakukan drag-and-drop grafis (<http://scratch.mit.edu>). Scratch tersedia dalam versi online dan dekstop, bahkan juga tersedia aplikasi scratch junior untuk versi android dan i-phone yang lebih memudahkan kita dalam berlatih. Dengan aplikasi scratch kita bisa mempelajari strategi-strategi penting untuk memecahkan masalah, mendesain karya, mengomunikasikan ide-ide dengan melakukan pemrograman. Sebenarnya orientasi pembelajaran pemrograman pada anak ini tidak hanya bertujuan untuk menghasilkan produk teknologi, namun juga untuk membentuk computational thinking skills sejak dini. Berikut contoh proses penyelesaian masalah yang bisa kita lakukan dengan aplikasi scratch :

1. Siswa akan menentukan masalah yang akan dipecahkan, misal : kita akan membuat games “Hungry Shark”. Lalu siswa akan menyusun analisis atau algoritma terhadap masalah tersebut. Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis dan logis (Budiyanto, 2017). Oleh karena itu langkah-langkah dalam algoritma harus logis dan dapat ditentukan nilai salah atau benarnya. Dalam dunia programming, menguasai algoritma menjadi hal yang penting karena algoritma adalah tumpuan untuk menyelesaikan sebuah persoalan. Pembuatan atau penulisan algoritma tidak tergantung pada bahasa pemrograman manapun. Cara menyajikan algoritma ada 3 yaitu Uraian deskriptif, Pseudo Code dan Flowchart. Flowchart adalah penggambaran secara grafik dari urutan prosedur suatu program. Flowchart dapat membantu kita memecahkan masalah ke dalam segmen-segmen yang lebih kecil. Berikut contoh flowchart dari games “Hungry Shark”:

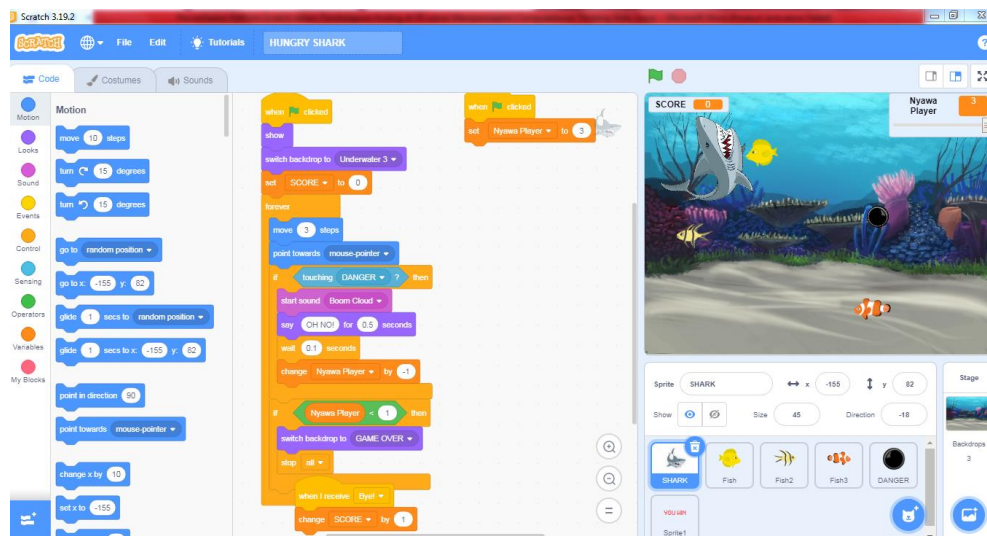


**Grafik 2. flowchart games “Hungry Shark”**

Jadi, ketika permainan “Hungry Shark” dimulai, shark bisa menyentuh ikan kecil atau bom. Apabila shark menyentuh ikan kecil, maka poin kita akan bertambah 1, jika tidak maka jumlah poin kita akan tetap (tidak bertambah/berkurang). Lalu, apabila poin berjumlah 25 artinya kita menang dan permainan pun selesai. Sedangkan apabila kita menyentuh bom, maka kesempatan bermain kita (nyawa) akan berkurang 1, jika tidak maka jumlah kesempatan bermain (nyawa) kita akan tetap (tidak bertambah/berkurang). Lalu, apabila kesempatan bermain (nyawa) kita berjumlah 0 artinya kita kalah dan permainan pun selesai. Dari sini kita bisa lihat bahwa algoritma yang telah kita buat dapat menjadi gambaran untuk melakukan kegiatan coding dengan menyusun bahasa pemrograman yaitu dengan cara menyusun balok coding yang tersedia pada scratch.

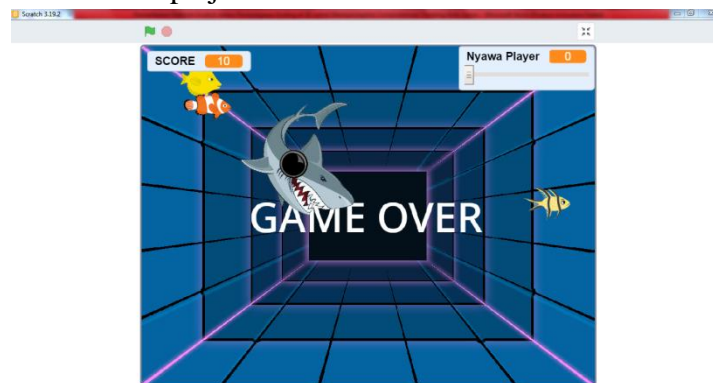
2. Dalam aplikasi scratch, bahasa pemrogramannya sudah tersedia. Siswa hanya perlu melakukan drag-and-drop untuk menyusun susunan-susunan balok coding yang benar agar program yang dibuat dapat berjalan. Di sini siswa perlu menganalisis apakah hubungan tiap balok kodingnya itu logis atau tidak. Berikut contoh proses coding

(menyusun bahasa pemrograman) pada permainan hungry shark yang sebelumnya telah kita buat algoritmanya :



**Gambar 1. Contoh Koding (Bahasa Pemrograman) pada scratch**

3. Tahap terakhir adalah tahap uji coba.



Setelah kegiatan koding selesai, tahap terakhir adalah uji coba. Pada tahap ini kita dapat mengetahui apakah susunan balok koding kita benar/salah, apakah pemrograman yang kita lakukan berhasil atau tidak. Pada gambar di atas menunjukkan bahwa skor/poin kita berjumlah 10. Hal ini karena kita berhasil menyentuh ikan kecil sebanyak 10 kali. Lalu, jumlah nyawa kita adalah 0. Hal ini karena kita sudah menyentuh bom sebanyak 3 kali. Maka kita kalah dan permainan pun selesai. Maka pemrograman yang sudah kita buat/lakukan berhasil dan kita telah menemukan pemecahan masalah (membuat permainan hungry shark).

Dari pemaparan proses penyelesaian masalah permainan "Hungry Shark" dengan aplikasi scratch di atas menunjukkan bahwa aplikasi scratch dapat meningkatkan :

1. Penyelesaian masalah terutama yang berhubungan dengan numerik. Hal ini ditunjukkan dengan terlibatnya fungsi aritmetika dalam penyusunan balok koding pada aplikasi Scratch.
2. Kecerdasan siswa. Dengan membuat karya dalam aplikasi Scratch siswa akan memperoleh kemampuan tingkatan tertinggi salah satunya dalam level kognitif yakni C6 (mencipta) dengan cara yang menyenangkan.
3. Siswa lebih cepat untuk memahami teknologi yang ada di sekitar. Hal ini dikarenakan siswa akan terbiasa menggunakan teknologi secara mendalam sehingga memungkinkan siswa mudah beradaptasi dengan teknologi baru dikemudian hari.
4. Melatih otak siswa agar terbiasa berpikir secara logis, terstruktur, dan kreatif.

Paparan didukung dengan review dari 20 pengguna Scratch yang 5 di antaranya merupakan pengajar sedangkan 14 lainnya merupakan pelajar. Adapun hasil review yang didapatkan adalah sebagai berikut :

**Tabel 1. Persepsi Pengguna Terhadap Pengaruh Scratch Pada Computational Thinking Skills**

Indikator	Ya	Tidak
Mengetahui tentang <i>computational thinking skills</i>	19	1
<i>Computational thinking skills</i> dapat dilatih melalui scratch	19	1
Scratch mudah digunakan untuk siswa SD	19	1
Scratch cocok dioperasikan untuk siswa SD	20	
Scratch menarik untuk diterapkan di sekolah dasar	20	
Keterampilan lain yang dapat dilatih melalui scratch	Ketelitian, kepekaan terhadap warna dan keterampilan berpikir kritis.	

Berdasarkan tabel di atas, tampak bahwa pada indikator pertama sebanyak 19 orang responden sudah mengetahui mengenai *computational thinking skills* dan 1 orang responden belum mengetahuinya. Pada indikator kedua sebanyak 19 orang responden beranggapan bahwa *computational thinking skills* dapat dilatih melalui scratch dan 1 orang responden tidak beranggapan demikian. Pada indikator ketiga sebanyak 19 orang responden merasa Scratch mudah digunakan untuk siswa SD dan 1 orang responden tidak

merasa demikian. Pada indikator keempat semua responden berpendapat bahwa Scratch cocok dioperasikan untuk siswa SD. Pada indikator kelima semua responden berpendapat bahwa Scratch menarik untuk diterapkan di sekolah dasar. Pada indikator keenam responden berpendapat bahwa ada keterampilan lain yang dapat dilatih melalui scratch, diantaranya : Ketelitian, kepekaan terhadap warna dan keterampilan berpikir kritis.

## **KESIMPULAN**

Computational thinking skills adalah kemampuan memecahkan masalah dengan cara penalaran dan analisa serta mengarahkan tindakan dengan langkah-langkah yang sistematis. Computational thinking skills ini sangat penting diterapkan pada siswa sejak dini untuk meningkatkan kecerdasan, membuat siswa lebih cepat untuk memahami teknologi yang ada di sekitar dan melatih otak siswa agar terbiasa berpikir secara logis, terstruktur, dan kreatif. Mengingat PISA 2021 pun akan menjadi PISA pertama yang mengevaluasi kemampuan berpikir komputasional siswa dengan mengintegrasikannya ke dalam sub-pengukuran bidang matematika. Salah satu cara menumbuhkan computational thinking skills khususnya siswa sekolah dasar adalah dengan menggunakan aplikasi Scratch. Scratch adalah bahasa pemrograman komputer untuk anak-anak yang dapat dilakukan dengan cara sederhana, yakni pengguna hanya melakukan drag-and-drop grafis.

Berdasarkan hasil penelitian disimpulkan bahwa melalui tahapan penyelesaian masalah yang dilakukan pada Scratch dapat meningkatkan :

1. Penyelesaian masalah terutama yang berhubungan dengan numerik. Hal ini ditunjukkan dengan terlibatnya fungsi aritmetika dalam penyusunan balok koding pada aplikasi Scratch.
2. Kecerdasan siswa. Dengan membuat karya dalam aplikasi Scratch siswa akan memperoleh kemampuan tingkatan tertinggi salah satunya dalam level kognitif yakni C6 (mencipta) dengan cara yang menyenangkan.
3. Siswa lebih cepat untuk memahami teknologi yang ada di sekitar. Hal ini dikarenakan siswa akan terbiasa menggunakan teknologi secara mendalam sehingga memungkinkan siswa mudah beradaptasi dengan teknologi baru dikemudian hari.
4. Melatih otak siswa agar terbiasa berpikir secara logis, terstruktur, dan kreatif.



Pemaparan di atas pun didukung dengan *review* dari 20 pengguna Scratch yang 5 di antaranya merupakan pengajar sedangkan 14 lainnya merupakan pelajar. Dari 20 responden dapat disimpulkan bahwa sebagian besar responden sudah mengetahui mengenai *computational thinking skills* dan beranggapan bahwa *computational thinking skills* dapat dilatih melalui scratch, responden juga merasa Scratch cocok, mudah dan menarik digunakan untuk siswa SD. Lalu responden juga berpendapat bahwa keterampilan lain yang dapat dilatih melalui scratch, diantaranya : Ketelitian, kepekaan terhadap warna dan keterampilan berpikir kritis.

## DAFTAR PUSTAKA

- Muhammad Zuhair Zahid. (2020, March 2). "Computational Thinking" Menyongsong PISA 2021. Retrieved July 11, 2021, from detiknews website: <https://news.detik.com/kolom/d-4922046/computational-thinking-menyongsong-pisa-2021>
- Moleong, L. J. A. Pendekatan dan Jenis Penelitian. *FAKULTAS PSIKOLOGI UNIVERSITAS ISLAM NEGERI MALANG*, 39.
- Nurhopipah, A., Nugroho, I. A., & Suhaman, J. (2021). PEMBELAJARAN PEMROGRAMAN BERBASIS PROYEK UNTUK MENGEMBANGKAN KEMAMPUAN COMPUTATIONAL THINKING ANAK. *JURNAL PENGABDIAN KEPADA MASYARAKAT*, 27(1), 6-13.
- Pratama, Y. (2019). *PERANCANGAN GAME COMPUTATIONAL THINKING UNTUK ANAK SEKOLAH DASAR* (Doctoral dissertation, UNIKA SOEGIJAPRANATA SEMARANG).
- Rosadi, M. E., Wagino, W., Alamsyah, N., Rasyidan, M., & Kurniawan, M. Y. (2020). Sosialisasi Computational Thinking untuk Guru-Guru di SDN Teluk dalam 3 Banjarmasin. *Jurnal SOLMA*, 9(1), 45-54.
- Saragih, R. R. (2016). Pemrograman dan bahasa pemrograman.
- Sholikhah, A. (2016). Statistik deskriptif dalam penelitian kualitatif. *KOMUNIKA: Jurnal Dakwah dan Komunikasi*, 10(2), 342-362.
- [scratch.mit.edu](https://scratch.mit.edu). Tentang Scartch. Diakses pada 11 Juli 2021 melalui <https://scratch.mit.edu/about>
- Zahid, M. Z., Dewi, N. R., Asih, T. S. N., Winarti, E. R., Putri, T. U. K., & Susilo, B. E. (2021, February). Scratch Coding for Kids: upaya memperkenalkan mathematical

thinking dan computational thinking pada siswa sekolah dasar. In *PRISMA, Prosiding Seminar Nasional Matematika* (Vol. 4, pp. 476-486).