

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan pustaka

Muhammad Anwar (2015) meneliti tentang “Teknologi WEBRTC untuk Bimbel *Online* dengan menggunakan teknologi WebRTC” yang diimplementasikan pada *interface Web* dengan bahasa pemrograman PHP.

Mohammad Vicky Agassi (2016) melakukan analisa terhadap protokol SRTP yang digunakan oleh WebRTC dalam mengenkripsi data mediastream, sehingga dapat diketahui apakah protokol SRTP mampu untuk mengamankan data mediastream dari penyadapan pihak ketiga.

Riski Marvianto Saputro (2017) menerapkan teknologi webrtc untuk membangun radio melalui jaringan internet dengan menambahkan fitur mengirim pesan (*chatting*) yang belum dimiliki oleh radio konvensional, sehingga pengguna dapat melakukan interaksi langsung kepada penyiar.

Pada skripsi yang ditulis oleh Nicholas (2014) dengan berisi mengenai bagaimana membuat chatroom menggunakan AJAX dan PHP berbasis web. Web ini bisa digunakan lebih dari 2 pengguna.

Pada skripsi yang ditulis Nisrina Luthfiati (2017) membuat aplikasi chat menggunakan teknologi websocket dan node js berbasis web. aplikasi ini menggunakan basis data berupa mongoDB. mengenai fungsi aplikasi ini digunakan sebagai sarana diskusi *online* yang dapat dimonitor oleh *super user*.

Penelitian yang diusulkan berupa pembuatan aplikasi *live editor text* dengan menggunakan teknologi webrtc yang diimplementasikan pada *interface web*. Perbandingan sistem dapat dilihat pada table 2.1

Tabel 2.1 Tabel perbandingan

Penulis	Objek	Teknologi	Bahasa Pemrograman	implementasi
Muhammad Anwar (2015)	Bimbingan Belajar Online	WebRTC	PHP, javascript	Menggunakan WebRTC yang menghasilkan aplikasi video streaming
Mohammad Vicky Agassi (2016)	Analisa Protokol SRTP (Secure Real Time Protocol) Pada WebRTC Dalam Upaya Menghindari Penyadapan Audio	WebRTC	Javascript	Menganalisa keamanan pada WebRTC khususnya pada protokol srtp
Riski Marvianto Saputro (2017)	APLIKASI WEB RADIO LIVE STREAMING MENGGUNAKAN TEKNOLOGI WEBRTC	WebRTC	Javascript	Memfaatkan WebRTC untuk membangun radio streaming melalui jaringan internet dengan menambahkan fitur mengirim pesan (chatting)
Nicholas (2012)	Pengembangan Aplikasi Chatroom	AJAX	Php	Membuat chatroom

	Menggunakan Ajax Dan Php Berbasis Web			menggunakan AJAX dan PHP berbasis web
Nisrina Luthfiati (2017)	Implementasi Websocket dan Nodejs Dalam Aplikasi Chatting Berbasis Web	Websocket, Node.js dan MongoDB	javascript	Membuat aplikasi chatting berbasis web menggunakan teknologi websocket
Usulan	Aplikasi live editor teks berbasis web	WebRTC	HTML dan Javascript	mengimplementasikan teknologi webRTC untuk membangun aplikasi editor teks yang bisa dibuat bersama-sama secara realtime

2.2 Dasar Teori

2.2.1 WebRTC

WebRTC adalah sekumpulan standar dari World Wide Web Consortium (WC3) dan *Internet Engineering Task Force* (IETF) yang diterapkan sejak awal 2011, dimana WebRTC dapat melakukan *real-time communication* (RTC) dalam web antar browser (Sam Dutton, 2014). WebRTC membuat pengguna dapat menambahkan komunikasi *real time* langsung dari *web browser* ke perangkat-perangkat lainnya. Bentuk komunikasi yang digunakan antara lain video, suara dan data langsung di dalam halaman *web browser* tanpa perlu memasang *plugin* apapun.

Komponen WebRTC yang dipakai dalam pembuatan *live editor text* meliputi :

1. RTCDataChannel API

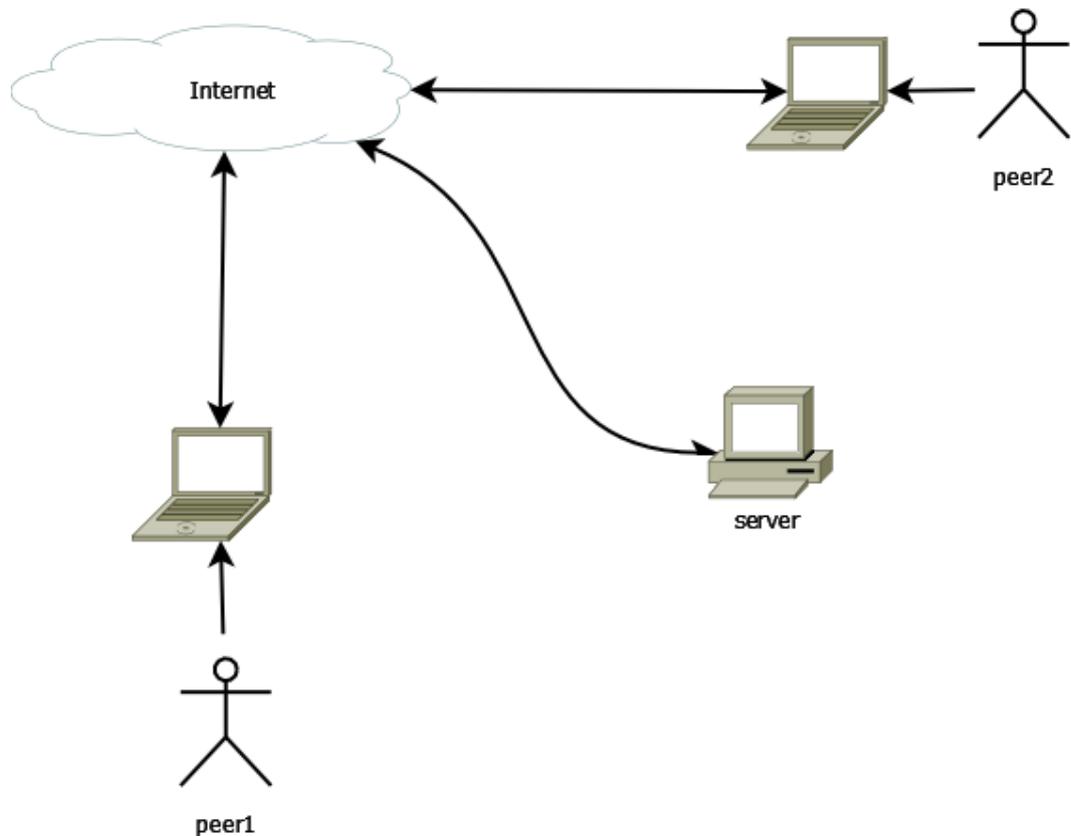
Membantu terjadinya proses pertukaran data antar peer, dengan tingkat latency yang rendah dan throughput yang tinggi. Jenis pertukaran data macam ini dapat sangat membantu dalam pengembangan aplikasi seperti : game, aplikasi *remote desktop*, *real-time text chat*, transfer file dan desentralisasi jaringan.

2. RTCPeerConnection

merupakan komponen WebRTC yang mengatur komunikasi data streaming diantara *peers* secara efisien dan stabil. *RTCPeerConnection* membantu menyederhanakan kompleksitas koneksi yang ada.

2.2.2 Live Editor Text

Live editor text adalah proses edit (dalam hal ini pengeditan teks) secara langsung/*realtime* dalam satu waktu yang bersamaan. Proses ini menggunakan koneksi internet sebagai media penghubung *realtimenya*. Ketika pengguna 1 menuliskan kata-kata maka saat itu juga tulisan dari pengguna 1 itu sendiri muncul di layar milik pengguna 2. Sebaliknya ketika pengguna 2 menuliskan kata-kata maka saat itu juga tulisan dari pengguna 2 itu sendiri muncul di layar milik pengguna 1.



Gambar 2.1 Arsitektur *Live Editor Text*

2.2.3 HTML

HTML adalah singkatan dari *Hypertext Markup Language*. Disebut *Hypertext* karena didalam HTML sebuah text biasa dapat berfungsi lain, kita dapat membuatnya menjadi link yang dapat berpindah dari satu halaman ke halaman lainnya hanya dengan mengklik text tersebut. kemampuan text inilah yang dinamakan hypertext, walaupun pada implementasinya nanti tidak hanya text yang dapat dijadikan link. Disebut *Markup Language* karena bahasa HTML menggunakan tanda (mark), untuk menandai bagian-bagian dari text. Misalnya, teks yang berada di antara tanda tertentu akan menjadi tebal, dan jika berada di antara tanda

lainnya akan tampak besar. Tanda ini dikenal sebagai HTML tag (duniaikom.com, 2017).

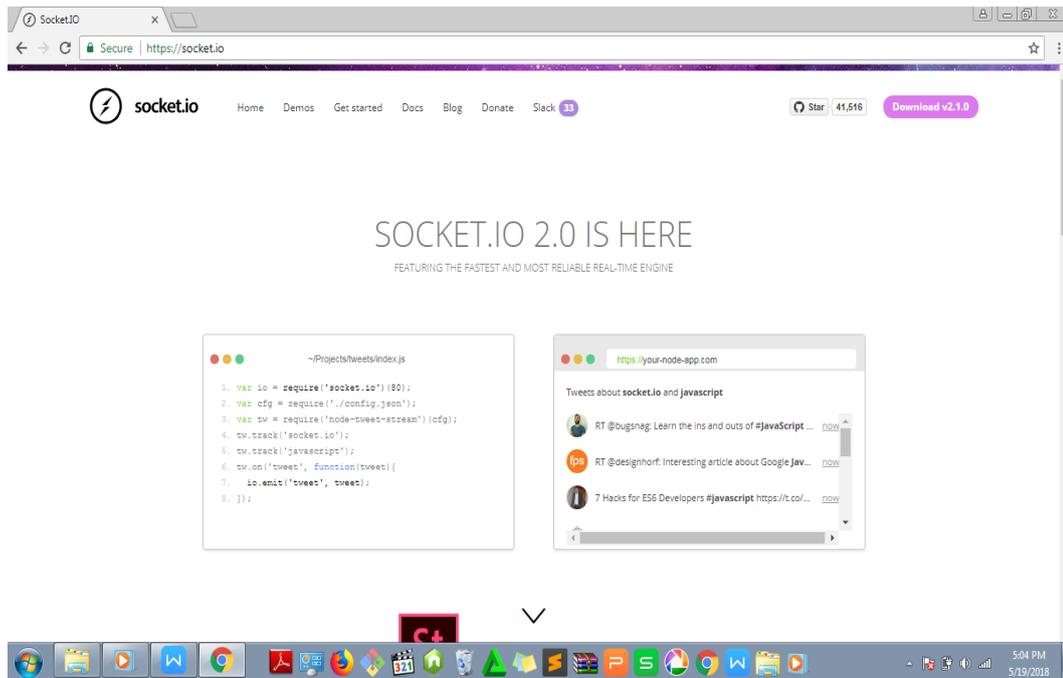
2.2.4 Javascript

Javascript adalah sebuah bahasa komputer atau kode pemrograman yang digunakan pada *website* agar *website* tersebut menjadi lebih interaktif dan dinamis. Javascript adalah jenis bahasa pemrograman *client side*. Penggunaan kode javascript pada sebuah website bersifat opsional, artinya tidak harus selalu ada. (devaradise.com, 2014). Javascript ini digunakan sebagai implementasi sistem

2.2.5 Socket io

Socket io merupakan sebuah *library* javascript yang membantu dalam pembuatan aplikasi web yang *realtime* lebih mudah. Dengan menggunakan socket io dapat menghubungkan antara *client* dan *server* dapat terjadi secara *birectional* (dua arah) yaitu dapat menghubungkan *client* dan *server* sehingga dapat berperan sebagai pengirim dan sekaligus penerima data. komponen yang terdapat pada socket.io terdiri dari dua bagian yaitu *client-site* yang berjalan pada browser dan *server-site* yang dapat digunakan sebagai modul untuk nodejs. Pada socket.io terdapat API, yang berfungsi untuk melakukan transaksi data. terdapat 2 metode penting yaitu `emit()` dan `on()`. `Emit()` berfungsi untuk mengirimkan sebuah pesan dari server ke klien dan dari klien ke server, saat data yang dikirimkan telah sampai pada tujuannya klien /server dapat menerima dengan

menggunakan method `on()`. Fungsi dari *method* `on()` untuk menerima data atau informasi yang telah diterima kemudian diproses oleh fungsi *handler* yang dibuat. (kursuswebsite.org)



Gambar 2.2 Socket IO

2.2.6 Froala editor

Froala Editor adalah *web editor* berbasis WYSIWYG (*what you see is what you get*). Ini adalah *platform-independent*, web berbasis JavaScript HTML WYSIWYG *editor* teks yang dibuat oleh Froala. (froala.com). *Web editor* ini yang digunakan pada sistem sebagai penampung teks yang diketik oleh pengguna.