PUSAT PRESTASI NASIONAL
KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

LKS SMK TINGKAT NASIONAL **28**

PANDUAN TEKNIS
PELAKSANAAN LKS SMK
TINGKAT NASIONAL XXVIII
TAHUN 2020

Teknik Desain Laman
Web Technologies

KATA PENGANTAR

Salah satu dari 4 pilar utama visi Indonesia tahun 2045 adalah pembangunan manusia dan penguasaan IPTEK (lmu Pengetahuan dan Teknologi), dengan peningkatan taraf Pendidikan rakyat Indonesia secara merata, peran kebudayaan dalam pembangunan, sumbangan IPTEK (lmu Pengetahuan dan Teknologi) dalam pembangunan, derajat kesehatan dan kualitas hidup rakyat, serta reformasi ketenagakerjaan. Sejalan dengan visi tersebut, dalam peningkatan pendidikan IPTEK (ilmu Pengetahuan dan Teknologi) merata pada era digitalisasi ini, siswa Sekolah Menengah Kejuruan (SMK) dituntut tidak saja harus menguasai penggunaan peralatan digital tetapi juga wajib menguasai softskill yang mumpuni.

Karena IPTEK dan komunikasi saling terkait dan tidak bisa dipisahkan, maka pada era digitalisasi disruptif, akan ada pekerjaan baru yang tercipta dan pekerjaan konvensional yang akan hilang. Untuk itu, siswa SMK harus senantiasa meningkatkan kualitas diri dan penguasaan keterampilan agar dapat memenuhi tuntutan pasar kerja, baik di masa kini maupun di masa yang belum kita prediksikan. Pekerjaan – pekerjaan yang selama ini dikerjakan yang sudah ada akan digantikan oleh sistim *Artificial Intelligence* (AI), otomatisasi atau robot yang dapat mengambil alih beberapa peran kerja manusia. Namun secanggih-canggihnya kemajuan IPTEK, hal yang pasti muskil digantikan oleh AI adalah *softskills* seperti Komunikasi & Empati, Berpikir Kritis, Kreatifitas, Strategi, Pengelolaan Teknologi, instalasi dan maintenance, keterampilan fisik, dan visi & imajinasi. Era digitalisasi maupun otomasi, dapat mengubah struktur ekonomi maupun tenaga kerja di Indonesia, kecuali beberapa pekerjaan yang sulit diotomasi misalnya kemampuan *softskills* (berinteraksi dengan orang lain dan keahlian khusus).

Lomba Kompetensi Siswa (LKS) SMK Tingkat Nasional XXVIII Tahun 2020 ini akan berbeda dengan LKS pada umumnya, dengan munculnya pandemi Covid-19 mendorong Indonesia untuk berubah dan tidak lagi menjalankan pola-pola yang lama. Seluruh lomba-lomba yang diselenggarakan oleh Pusat Prestasi Nasional dilakukan secara daring dengan memperhatikan protokol kesehatan *Covid-19*. Sisi baik dari tantangan ini adalah siswa SMK diajak untuk bersahabat dan berkolaborasi dengan teknologi daring. Pusat Prestasi Nasional melakukan pembaharuan dengan melaksanakan LKS 2020 secara daring. LKS Tingkat Nasional Tahun 2020 melombakan sebanyak 42 bidang lomba. Diharapkan pada masa pandemi Covid-19 tidak mengurangi semangat siswa untuk berprestasi.

Sehubungan dengan hal tersebut, Pusat Prestasi Nasional, Sekretariat Jenderal, Kementerian Pendidikan dan Kebudayaan ikut mendukung pengembangan kualitas SMK dalam mengikuti perkembangan IPTEK dan memenuhi Visi Indonesia 2045. LKS Tingkat Nasional Tahun 2020 adalah salah satu kegiatan untuk mendorong semangat berprestasi

peserta didik SMK yang diadakan setiap tahun dan sebagai upaya mempromosikan lulusan SMK kepada dunia usaha/dunia industri serta pemangku kepentingan lainnya.

Panduan Teknis LKS SMK Tingkat Nasional XXVIII Tahun 2020 Daring merupakan dokumen pendukung pelaksanaan LKS demi tercapainya kegiatan agar berjalan dengan baik dan dapat memberikan informasi kepada semua pihak yang ikut berpartisipasi dalam pelaksanaan LKS.

Dalam kesempatan ini disampaikan ucapan terimakasih kepada semua pihak yang telah mendukung dalam penyusunan Panduan Teknis pelaksanaan LKS SMK Tingkat Nasional XXVIII Tahun 2020.

Plt. Kepala Pusat Prestasi Nasional

Asep Sukmayadi, S.IP., M.Si
NIP. 197206062006041001

DAFTAR ISI

A   Pendahuluan

A.1   Nama dan Deskripsi Lomba

A.1.1   Nama Bidang Lomba

Web Technologies

A.1.2   Deskripsi Lomba

Web Technologies mencakup berbagai keterampilan dan disiplin dalam produksi dan pemeliharaan situs web. Keterampilan yang dibutuhkan *developer web* sangat beragam, seringkali sulit bagi *developer* untuk unggul dalam semua aspek. Akibatnya, tim dapat mengikuti proses desain *web*, dengan setiap anggota tim memiliki kekuatan, spesialisasi, dan peran masing-masing dalam proses pengembangan.

A.1.3   Isi Deskripsi Teknis

Design melibatkan penerapan solusi spesifik yang mengikuti aturan dan tujuan bisnis yang dideskripsikan oleh klien. *Web Designer* mengembangkan hubungan profesional dengan klien mereka, berinteraksi untuk mengembangkan pemahaman mendalam tentang persyaratan, dan mengubahnya menjadi spesifikasi situs *web*. Desain dan kemampuan komunikasi yang kuat, ditambah dengan teknik penelitian dan pemahaman khalayak target, pasar dan tren, akan memastikan kepuasan dan kredibilitas klien awal untuk *Web Designer*. Setelah menyelesaikan perencanaan dan perancangan situs web, *Web Designer* kemudian mengintegrasikan situs *web* dengan alat dan platform pihak ketiga. Selama proses pengembangan *web designer* menerapkan desain, menggunakan keterampilan pemrograman mereka untuk menciptakan fungsionalitas dinamis, tes, dan debug situs web dengan menggunakan berbagai perangkat. Tren saat ini adalah untuk juga mengintegrasikan situs web dengan media sosial untuk memanfaatkan platform pemasaran online yang ada. Semua keterampilan ini mungkin berlaku sama untuk desain ulang atau upgrade dari situs web yang ada.

Perancang Web berperforma tinggi mungkin memiliki keahlian *web-related* yang luas atau khusus. Mereka harus memahami nilai artistik, memiliki kemampuan mendesain antarmuka pengguna yang solid, keterampilan pemrograman, dan

bertanggung jawab secara pribadi karena selalu berada di garis depan tren dan teknologi *web*. Mereka juga harus responsif terhadap klien dan memiliki kemampuan untuk bekerja dalam tim dan kelompok terstruktur dan tidak terstruktur. Kualitas ini memungkinkan *Web Designer* untuk berkontribusi dan memanfaatkan aspek teknologi komunikasi modern yang berkembang pesat ini.

A.2 Dokumen Terkait

Dokumen ini hanya berisi informasi tentang aspek teknis keterampilan, dokumen lain yang juga harus dipelajari adalah:

1. Pendoman lomba,
2. Informasi di website panitia:
   a. Kisi-kisi soal LKS
   b. Rencana Kerja
   c. Form Kebutuhan Bahan
   d. Lembar Ceklis Kebutuhan Bahan

B   Standar Kompetensi Bidang Lomba

B.1 Ketentuan Umum

Spesifikasi Kompetensi adalah rumusan target kompetensi yang akan dilombakan. Target kompetensi dirumuskan berdasarkan situasi dunia kerja atau industri dengan tetap memperhatikan kurikulum SMK.

LKS mengukur pengetahuan dan pemahaman melalui penampilan/unjuk kerja.Proyek uji, skema penilaian dan bobot masing-masing modul proyek uji dikembangkan berdasarkan spesifikasi kompetensi LKS-SMK.

B.2   Spesifikasi Kompetensi LKS SMK

No.	Standar Kompetensi	Tujuan	Kriteria	Praktikum /Modul	Presentase
1	Manajemen dan Organisasi Kerja			Modul A, B C	6
	Mengetahui dan memahami: • Prinsip dan praktik yang memungkinkan kerja tim produktif • Prinsip dan perilaku system • Aspek sistem yang berkontribusi terhadap produk, strategi yang berkelanjutan dan praktik • Bagaimana cara berinisiatif dan giat untuk mengidentifikasi, menganalisis dan mengevaluasi informasi dari berbagai sumber • Identifikasi beberapa solusi untuk masalah dan menawarkan opsi terhadap waktu, anggaran, dan kendala lainnya.				

	Dapat melaksanakan:				
	• *Troubleshoot* masalah desain dan pengembangan *web* secara umum				
	• Membatasi waktu dan tenggat waktu				
	• *Debug* dan *Error handling*				
	• Menggunakan komputer atau perangkat dan berbagai paket perangkat lunak				
	• Terapkan teknik dan keterampilan penelitian untuk selalu mengikuti perkembangan terbaru berdasarkan pedoman industri				
	• Merencanakan jadwal produksi setiap hari sesuai waktu yang tersedia				
	• Menyertakan gambar terkait, font, file asli dan format file produksi saat pengarsipan				
	• Menggunakan sistem *version control*				

2.	Komunikasi dan Keterampilan Interpersonal			Modul A, B, C	6
	Mengetahui dan memahami: • Bagaimana mengatasi masalah komunikasi termasuk mengidentifikasi masalah, penelitian, analisis, pembuatan solusi, prototyping, user pengujian dan evaluasi hasil • Konsep dan teknik desain termasuk wireframing, storyboard, dan membuat flowchart • Konsep dan teknik perancangan perangkat lunak termasuk flowchart dan ER diagram Dapat melaksanakan: • Membuat, menganalisa, dan mengembangkan respon visual terhadap komunikasi masalah, termasuk pengertian hirarki, tipografi,				

	estetika, dan komposisi • Membuat, manipulasi dan optimalkan gambar untuk internet • Identifikasi target pasar dan buat konsep untuk desain • Buat desain responsif yang berfungsi dengan benar pada beberapa layar resolusi dan / atau perangkat • Mengubah ide menjadi desain yang estetis dan kreatif • Konsep konsep kritik, pilihan warna dan tipografi				
3.	*Design Website*			Modul A, B, C	22
	Mengetahui dan memahami: • Cara mengikuti prinsip dan pola desain agar bisa memproduksi desain yang estetis dan kreatif • Isu yang berkaitan dengan kognitif, sosial, budaya, teknologi dan konteks ekonomi untuk desain				

	- Cara membuat dan mengadaptasi grafis untuk web - Target pasar yang berbeda dan elemen desain yang memuaskan pasar - Protokol untuk menjaga identitas korporat, brand dan gaya - Keterbatasan perangkat dan resolusi layar Internet Dapat melaksanakan: - Membuat, menganalisa, dan mengembangkan respon visual terhadap komunikasi masalah, termasuk pengertian hirarki, tipografi, estetika, dan komposisi - Buat, manipulasi dan optimalkan gambar untuk internet - Identifikasi target market dan buat konsep untuk desain - Buat desain responsif				

	yang berfungsi dengan benar pada beberapa layar resolusi dan / atau perangkat • Mengubah ide menjadi desain yang estetis dan kreatif • Konsep konsep kritik, pilihan warna dan tipografi				
4.	Layout Website			Module A, B	22
	• Mengetahui dan memahami: Standar *World Wide Web Consortium* (W3C) untuk HTML dan CSS • Metode penentuan posisi dan tata letak • *Usability* dan desain interaksi • Aksesibilitas dan komunikasi bagi pengguna dengan kebutuhan khusus • *Cross browser* kompatibilitas • *Search Engine Optimization* (SEO)				

	• Cara menanamkan dan mengintegrasikan animasi, audio dan video bila dibutuhkan Dapat melaksanakan: • Buat kode yang sesuai dan validasikan dengan standar W3C • Buat situs web yang dapat diakses dan bermanfaat untuk berbagai perangkat dan layar resolusi • Gunakan CSS atau file eksternal lainnya untuk memodifikasi tampilan *website* • Gunakan CSS pre / *post-processors* • Buat dan perbarui situs web untuk pengalaman pengguna dan untuk membantu pencarian kinerja mesin				
5.	*Client Side Development*			Modul A	22
	Mengetahui dan memahami: • JavaScript • Bagaimana				

	mengintegrasikan *library*, *framework* dan sistem atau fitur lainnya dengan JavaScript Dapat melaksanakan: • Membuat animasi dan fungsionalitas situs *web* untuk membantu dalam konteks • Penjelasan dan menambahkan daya tarik visual • Membuat dan memperbarui kode JavaScript untuk meningkatkan fungsionalitas situs *web*, kegunaan dan estetika • Memanipulasi data dan media khusus dengan JavaScript • Membuat kode JavaScript modular dan dapat digunakan kembali				
6.	*Server Side Development*			Modul C	14
	Mengetahui dan memahami:				

	- PHP berorientasi objek *Open source libraries* dan *framework* - Bagaimana merancang dan mengimplementasikan database dengan MySQL - FTP (File Transfer Protocol) hubungan server dan klien dan perangkat lunak paket. - Bagaimana mengelola pertukaran data antara *server* dan sistem *client* - Pola perancangan perangkat lunak (misal: MVC (Model View Controller)) - Keamanan aplikasi web Dapat melaksanakan: - Memanipulasi data dengan memanfaatkan keterampilan pemrograman - Melindungi website dari eksploitasi keamanan - Integrasikan dengan				

	kode yang ada dengan API (*Application Programming Interface*), perpustakaan dan kerangka kerja				
7.	Content Management Systems			Modul B	8
	Mengetahui dan memahami: • Manfaat dan keterbatasan Sistem Manajemen Konten open source • Cara mencari, memilih dan menerapkan plugin / modul yang sesuai • Bagaimana menerapkan fungsi sisi klien ke situs web CMS • Pahami kebutuhan akan *maintenance* dan *update plugin* CMS dan modul untuk keamanan Dapat melaksanakan: • Instal, mengkonfigurasi dan perbarui CMS • Instal, mengkonfigurasi dan update plugin / modul CMS • Buat tema / template				

	khusus untuk CMS • Buat plugin / modul khusus				
	Jumlah				100%

C   Sistem Penilaian

C.1   Petunjuk Umum

Penilaian LKS-SMK menggunakan ketentuan yang telah ditetapkan panitia. menggunakan dua jenis, yaitu subyektif dan obyektif. Penilaian subyektif dilakukan dengan cara pengamatan proses maupun hasil. Untuk memudahkan justifikasi disediakan kriteria penilaian. Sedangkan penilaian obyektif didasarkan pada pengukuran kriteria.

C.1.1   Skema Penilaian

No.	Modul	Kriteria/Sub-Kriteria	Total
1	A	*Speed Test*	10
2	B	*CMS*	30
3	C	*Client Side*	30
4	D	*Server*	30
Total			100

D   Test Project

D.1   Petunjuk Umum

Proyek uji / *Test Project* dikembangkan untuk mengukur seluruh spesifikasi kompetensi LKS-SMK secara daring.

Tujuan penyusunan proyek uji adalah untuk penilaian pencapaian spesifikasi kompetensi LKS-SMK. (Projek Uji dibuat pada dokumen terpisah)

Tes Projek akan berubah minimal 30% dari kisi-kisi yang sudah diberikan

Aturan khusus keterampilan sudah ada pada Teknikal Deskripsi ini. Mungkin akan sedikit berbeda dengan dunia kerja sebenarnya dikarenakan memang aturan ini

dibuat untuk kepentingan keterampilan kompetisi dalam kondisi *Covid-19*. Termasuk juga tidak ada batasan untuk peralatan yang digunakan, prosedur dan alur kerja, serta pengelolaan dokumen dan distribusi

D.2 Kriteria Penilaian

Kriteria penilaian adalah hal utama dalam skema penilaian yang ditentukan berdasarkan proyek uji. Bobot masing-masing kriteria penilaian menyesuaikan dengan spesifikasi kompetensi LKS yang ditetapkan. Kriteria penilaian dikembangkan sesuai kepentingan proyek uji.

Modul	Deskripsi	Hari	Score	Jam
A	*Speed Test*	1	10	2,5
B	*CMS*	1	30	2,5
C	*Client Side*	2	30	5
D	*Server*	3	30	5

D.2.1 Persyaratan Proyek Uji

Skema penilaian menjelaskan tentang aturan dan bagian yang akan dinilai dalam lomba melalui proyek uji yang dikerjakan peserta serta proses penilaian. Skema penilaian dalam LKS-SMK dipergunakan untuk mengukur keterampilan peserta dalam mengerjakan proyek uji. Aspek penilaian dikembangkan berdasarkan spesifikasi kompetensi LKS-SMK dan pembobotan yang telah ditetapkan. **Test Project akan diberikan saat lomba.**

D.3 Sub Kriteria

Modul		Sub Kriteria
A	A1	*Design*
	A2	*Layout*
	A3	*Front-end*
	A4	*Back-end*
	A5	CMS
B	B1	Membuat *Style guide*
	B2	CMS *Element* sesuai soal
	B3	Kesesuaian tema
	B4	*Role* pada CMS
	B5	Desain responsif
C	C1	*Client side general*
	C2	Membuat *menu game*
	C3	Membuat *game*
	C4	Membuat *highscore*
D	D1	*Server* API untuk CRUD secara *general*
	D2	*Front End*

D.4   Aspek

Setiap kriteria dirumuskan dalam aspek penilaian yang memungkinkan diamati atau diukur, meliputi:

Sub Kriteria	Deskripsi
Speed Test	
	Mengerjakan *task-task* kecil dalam waktu singkat
CMS	
	Style guide dibuat dan sesuai dengan soal
	Membuat element CMS yang diminta soal
	Tema sesuai dengan soal
	Membuat *role* di CMS dan memiliki kewenangan yang berbeda
	Desain sesuai dengan ukuran yang dibuat pada *style guide*
Client Side	
	Client dibuat sesuai spesifikasi soal
	Membuat game menu dan dapat digunakan
	Game dibuat sesuai soal dengan fungsionalitas yang lengkap
	Client dapat menghitung highscore
Server	
	Server me-*return response* status sesuai soal
	Membuat *front end* sesuai soal

D.5   Penilaian

D.5.1   Penilaian Subjektif

Penilaian subyektif dilakukan untuk proses kerja dan hasil kerja yang berdasarkan pengamatan atau jastifikasi juri. Penilaian subyektif memerlukan kriteria (rubrik) untuk membantu proses penilaian.

Skala jastifikasi:

0: Tidak melakukan

1: dibawah rata-rata performa industri

2: diatas rata-rata performa industri

3: Sempurna

D.5.2   Penilaian Objektif

Penilaian obyektif dilakukan oleh minimal dua juri. Penilaian hanya memberikan angka 1 bila sesuai ukuran dan toleransi dan 0 bila tidak sesuai.

D.6   Komposisi Penilaian Subjektif dan Objektif

No	Module	Kriteria/Sub-kriteria	Subyektif	Obyektif	Total Akumulasi
1	A	*Speed Test*	0	10	10
2	B	CMS	14,5	15,5	30
3	C	*Client Side*	5,5	24,5	30
4	D	*Server*	3	27	30

*) jumlah item yang dinilai

D.7   Keseluruhan Aspek

Sub Kriteria	Deskripsi	Subjektif	Objektif	Total
Speed Test		0	10	10
A1	*Design*			
A2	*Layout*			
A3	*Front-end*			

A4	*Back-end*			
A5	CMS			
CMS		14,5	15,5	30
B1	Membuat *Style guide*			
B2	CMS Element sesuai soal			
B3	Kesesuaian tema			
B4	*Role* di CMS			
B5	Desain responsif			
Client Side		5,5	24,5	30
C1	*Client side general*			
C2	Membuat *menu game*			
C3	Membuat *game*			
C4	Membuat *highscore*			
Server Side		3	27	30
D1	*Server* API untuk CRUD secara *general*			
D2	*Front End*			
Total				100

D.8   Prosedur Asesmen

Modul	Deskripsi	Hari
A	*Speed Test*	1
B	CMS	2
C	*Client Side*	3
D	*Server*	3

E   Alat

E.1   Ketentuan Umum

Alat dan yang telah disedikan oleh panitia tidak dapat digantikan dengan alat dan bahan yang dibawa oleh peserta kecuali panitia meminta peserta untuk menyiapkan sesui dengan ketentuan yang sudah di tetapkan.

Peserta diberikan waktu familiarisasi fasilitas lomba sebelum lomba dimulai (maksimal 2 jam).

E.1.1   Daftar Sarana dan Prasarana

Demi lancarnya perlombaan, panitia diharapkan menyediakan hal berikut:

1. Komputer (spesifikasi terlampir), keyboard, mouse, meja, kursi untuk digunakan oleh peserta.
2. Koneksi internet untuk *Push Git* (upload jawaban). Internet tetap dinyalakan saat lomba.
3. Area lomba sesuai Pedoman LKS daring poin "E. Tata Tertib dan Teknis Pelaksanaan Lomba"

Alat dan yang telah disediakan oleh panitia tidak dapat digantikan dengan alat dan bahan yang dibawa oleh peserta.

Peserta diberikan waktu familiarisasi fasilitas lomba sebelum lomba dimulai.

E.1.2 Daftar Alat para Peserta

Alat yang dipersiapkan oleh peserta meliputi:

No	Tool / Equipment	Keterangan
Perlengkapan Safety		
1	Komputer Peserta	Monitor LCD 24 Inch, Intel Processor i3 seri 4 atau setara, RAM 4 GB, HD 500 GB, Keyboard, Mouse, OS Windows 10
2	*Internet*	*Unlimited* direkomendasikan
3	*Software*	XAMPP 7.2.11 (with PHP 7.2.11)
		Adobe Illustrator & Photoshop CC
		Sublime Text Editor
		Notepad++
		Brackets
		PHPStorm
		MySQL Workbench
		FileZilla
		Composer
		Node.js
4	*Smartphone* dengan kamera	Sesuai pedoman; dilengkapi dengan *tripod*. Menggunakan aplikasi Zoom. *Smartphone* tidak boleh mati karena baterai habis atau putus koneksi.

E.1.3 Alat dan bahan yang dilarang digunakan

Tidak diperbolehkan untuk menggunakan alat komunikasi seperti telepon genggam/smartphone selain untuk kebutuhan video conference. Dalam ruangan yang digunakan untuk lomba, hanya diperbolehkan peserta dan panitia jika ada keperluan teknis.
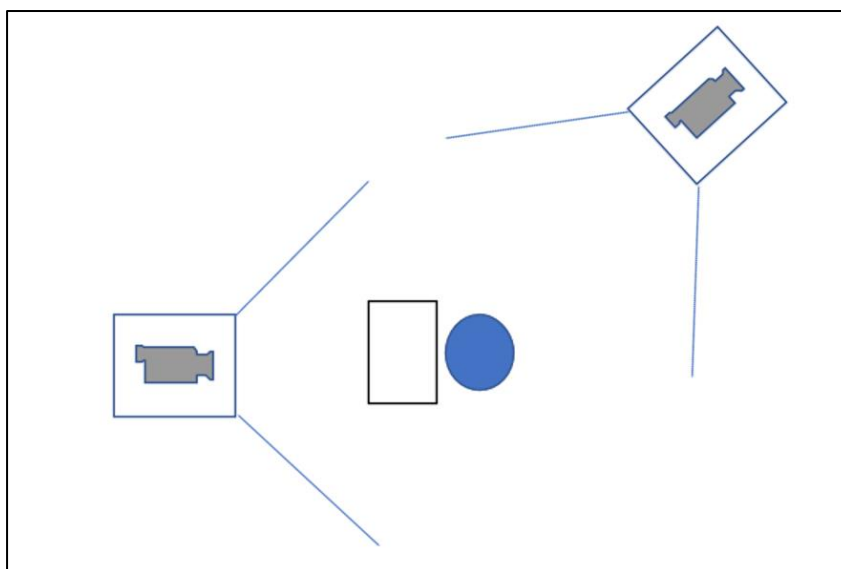
F   Bahan

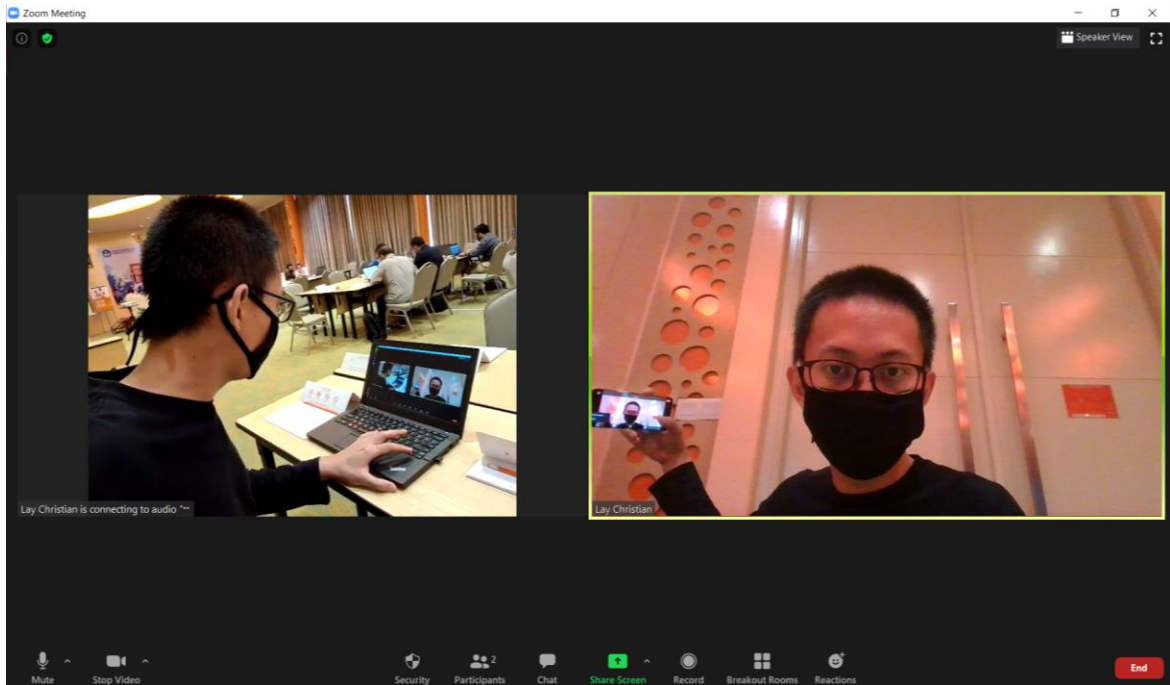(Dilampirkan)

F.1   Bahan dan perakitan

G   Bahan Penunjang

G.1   Bahan penunjang Lomba Sebagai Referensi para Peserta

H   *Layout*

H.1   *Layout*

Gambar 1 Layout perlombaan tiap peserta

Gambar 2 Contoh layout

Kamera menggunakan kamera *smartphone* sebanyak 2 buah. Kamera depan dapat digantikan *webcam laptop*. Diperlukan *tripod* untuk dapat menahan *smartphone* agar dapat memberikan gambar yang baik. Aplikasi yang digunakan untuk *video conference* adalah Zoom.

.

H.2   Tabel Kebutuhan Bahan untuk Layout

No	Tool / Equipment	Quantity	Satuan	Gambar
Material lay out				
1				

Note: Jika ada

I   Jadwal Bidang Lomba

Waktu		Kegiatan	Keterangan	
Hari C -1				
10.00 - 12.00	120'	*Briefing* Lomba dan soal		
Hari C1				
7.45 – 08.00	15'	*Briefing* Soal		
08.00 – 10.30	150'	Mengerjakan Modul A		
11.00 – 13.30	150'	Mengerjakan Modul B		
Hari C2				
7.45 – 08.00	15'	*Briefing* Soal		
08.00 – 13.00	300'	Mengerjakan Modul C		
Hari C3				
7.45 – 08.00	15'	*Briefing* Soal		
08.00 – 13.00	300'	Mengerjakan Modul C		

J   Kebutuhan lain dan spesifikasinya

J.1   Kebutuhan Juri untuk Menilai

No	Tool / Equipment	Quantity	Satuan	Gambar
Untuk Juri (Penilaian) (bisa sewa atau pinjam dari sekolahan)				
1				

J.2   Kebutuhan Perlombaan

No	Tool / Equipment	Quantity	Satuan	Gambar
1				

## K. Kisi-Kisi

SPEEDTEST

Task 1: Cube Rotation
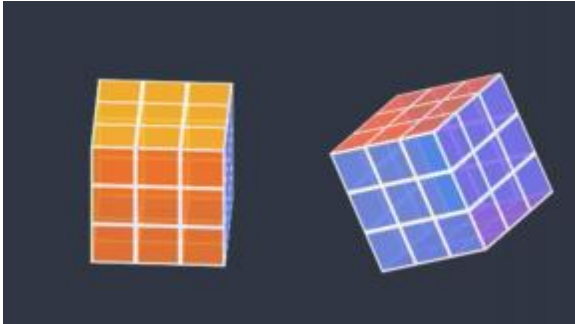
We want to create the following cube rotation effect.
1. You should create two cubes using the index.html and the cubes should be rotated same as video.mp4.
2. Left cube should be rotated left to right.
3. Right cube should be rotated up to down.
4. color codes
   - front - #fa5252
   - back - #f76707
   - right - #12b886
   - left - #4c6ef5
   - top - #fab005
   - bottom - #7950f2
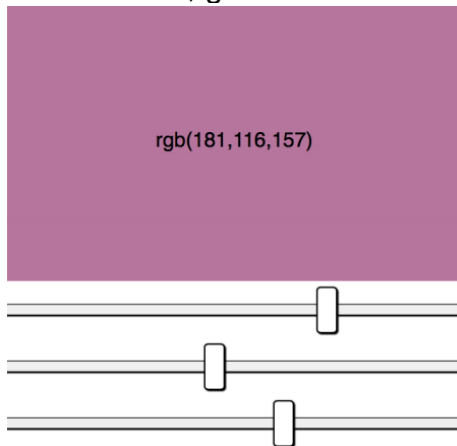5. You can only use style.css for the Cube Animation, JavaScript is not allowed.

Task 2: CSS Grid
Please implement the following layout in CSS grid.
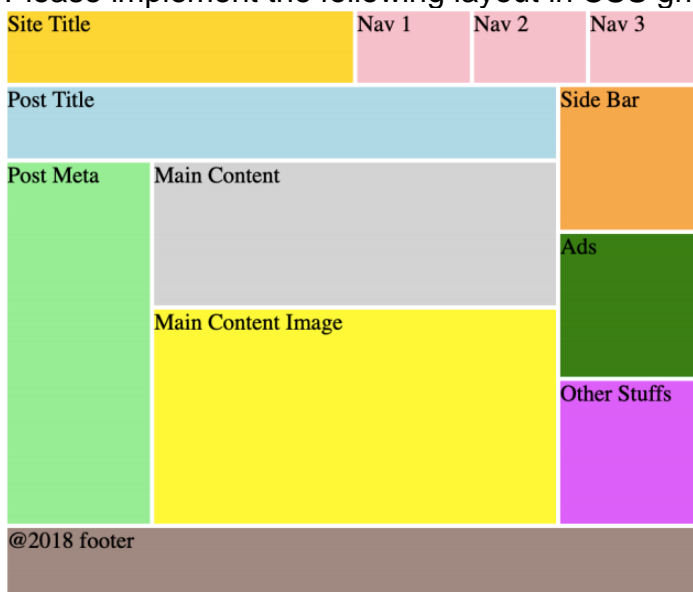
Task 3: RGB Slider

Please implement an RGB slider as following. There are 3 sliders for adjusting values of red, green and blue.

Task 4: PHP Array
Given the code with some PHP arrays in media files, please implement a PHP function that compare the given two arrays and return a new array that contains the common elements from both given arrays.

Task 5: Installing Wordpress
Now a client needs a wordpress blog to go up and live very quickly. They don't care about the layout so default theme should work. But they want to have a "Steelblue" top bar and "Light Cyan" background color on their blog. Please install a new wordpress website with the following requirement: • Title: Handy Cafe • Sub-title: A cafe that you never forget. • Top color bar on all page: 5 px steel blue • Background color on all page: light cyan • Categories: products, closing time, events • There are menus on home page that links to posts listing of the mentioned categories.

MODULE CLIENT SIDE

CONTENTS

This module has the following files:

1. MODULE_CLIENT_SIDE.doc
2. MODULE_CLIENT_SIDE_MEDIA.zip

INTRODUCTION

In recent years internet has become our basic daily source and needs, enabling the dissemination of information in an inexhaustible content and interaction. Games uses has gained a promined role in nowadays computer usage, allowing people to get access to entertainment from any places.

You are asked to develop game called **Flipping** using HTML and CSS and develop client-side programming using JavaScript and its open source libraries. Some media files are available to you in a zip file. You can create more media and modify anything in the media if you want.

Your game needs to be developed in a tablet resolution (960x600 pixels). In bigger resolution, the game must be centred in the screen both horizontally and vertically.

DESCRIPTION OF PROJECT AND TASKS

This is a module of 5 hours. Your first 2.5 hours must be used to create the design of the game in three PNG images and the initial layout using HTML/CSS. Your layout should follow the design that you created. The final 2.5 hours you will create the functionality of game using JavaScript that allows the game to work correctly in different web browsers.

Flipping game board are described below:

1. Board consist of **8x8 board** and 64 discs, 32 black and 32 white.

2. **Place 2 black and 2 white discs at the centre of the board**. Arrange the discs with matching colour diagonal to each other.

3. Total discs form each colour.

FIRST 2.5 HOURS – DESIGN AND INITIAL LAYOUT:

1. **Deliver at least 3 PNG image files that present:**
   1.1. Game Instructions: The first screen of the game presents the instructions to the player, a text field for player's name, and the "Colour" buttons for choosing color. The instructions for the game are included in the media files. The "Colour" button should be disabled if the text field is empty.
   1.2. Game board layout: It must present all elements described above in the game screen.
   1.3. Game Icon which represent the game immediately.

2. **Develop the initial markup (HTML + CSS) of your game application.** When the layout is accessed the game is presented to the user with the game instructions and the button "Start Game". The instructions must be presented in an animated way.

3. **"Colour" buttons must have active and hover effects.** The background of the buttons in hover state must be: #e2e9fc for white and #272729 for black. The active state must use #0069d9.

4. **The HTML and CSS** code must be valid in the W3C standards for HTML5 and CSS3 rules.

FINAL 2.5 HOURS – GAME FUNCTIONALITIES:

1. **Black always goes first in Flipping**. Player can choose between white or black; opposite colour will be run by computer.
2. **Pressing the "Colour" button in the initial screen will begin** the countdown from 3 before the game starts. Player cannot do anything in this state. After the countdown reaches 0, player can start playing the game with the 8x8 board.
3. **Game will shows every possible legal move suggestion** so it would be easier for players to make decisions. The suggested board should be highlighted.
4. **Player can click any suggested legal moves** to choose it. The chosen spot should be animated to show the board are chosen. If Player click on board without suggestion moves, animate and highlight all possible legal move.
5. Disc with colour based on player's colour will **appear** on spot player had chosen in advance.
6. **Place disc in a spot that surrounds an opponent's disc**. This is also known as "outflanking". A "row" consists of one or more discs that form a line horizontally, vertically, or diagonally.
    6.1. For example, if the opponent has a disc next to 1 of your discs in a vertical row, then place a disc on the open side of their disc in the same row to outflank your opponent's disc
7. **Flip the outflanked disc to its opposite side.** Once a disc is outflanked, flip it over to the opposite colour. However, the same disc may be turned over again if it is part of a row that is outflanked.
8. **Create score counters** which will count every disc on each colour in order to determine who will win at the end of the game.
9. **Pass the turn to your opponent to continue playing.** Each Player goal is to place a disc in a spot that outflanks at least 1 of the other player's discs.
10. **Continue taking turns placing discs until a legal move isn't possible.** Always place discs in a position where they can outflank a row of the opponent's discs. If this isn't possible, you must forfeit your turn until you can perform a legal move. If neither player can perform a legal move, then the game is over.
11. **Save the current score as high score in the local storage** when the game is over (whether the black or white won) with higher score than the current score.
12. **Use your talent to increase the usability of the game** as much as possible to permit a better experience for the user.
13. **There must be an option to disable/enable game sounds.** If the sound is disabled, none of sounds should be played. If sound is enabled all sounds must be played.
14. **Your game should work without JavaScript errors** or messages shown in the browser console.
15. **Maintain your HTML/CSS and JavaScript code organized and clean to facilitate future maintenance.** Use correct indentation and comments. Use meaningful variable names and document your code as much as possible so another developer would be able to modify your work in the future
16. The game needs to work correctly in **Google Chrome**

INSTRUCTIONS TO THE COMPETITOR

1. The media files are available in the ZIP file. You can modify the supplied files and create new media files to ensure the correct functionality and improve the application.

2. Save your design files in a folder call "**XX_CLIENT_SIDE_MODULE/design**" where XX is your computer number.

3. You should create additional images for each of the requested resolution to highlight hidden elements, animations, interactions, or any additional information that will assist in the presentation of the game design.

4. Additional file names

    a. Instructions: XX_instructions_2.png, XX_instructions_3.png

b. Game board: XX_game_board_2.png, XX_game_board_3.png

5. Save any image source files to a folder named "**source**" inside the "**XX_CLIENT_SIDE_MODULE/design**" folder. The source files are the files that contain the layers, development files, ie .psd, .ai, .svg, .jpg.

6. Save the working game to the directory on the server named "**XX_CLIENT_SIDE_MODULE**". Be sure that your main file is called index.html.

7. You are responsible for the time management in your development. If you finalize some tasks you can continue to other tasks. The initial 2.5 hours only define what will be evaluated first.
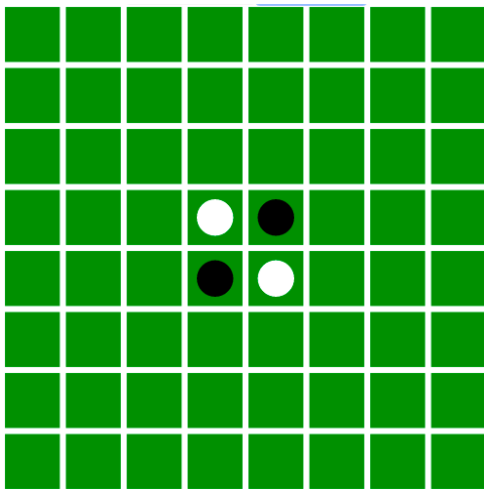
EXAMPLE
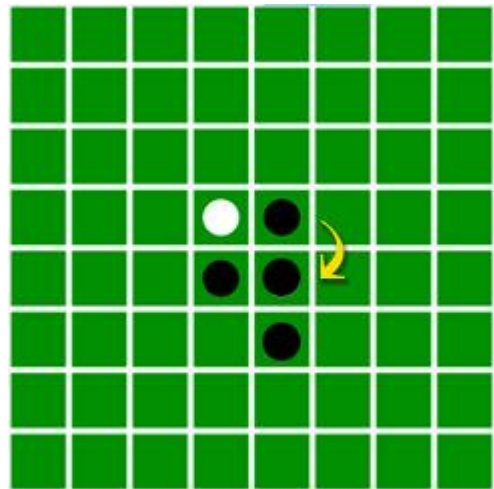

Image 1 Shows initial position Flipping Game
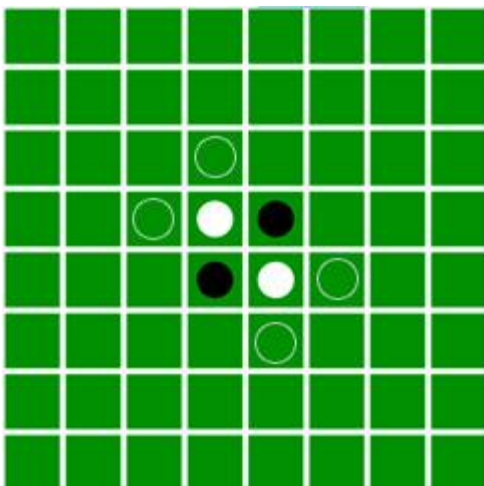

Image 3 Black Outflank White

Image 2 Legal Move Suggestion

MODULE CMS

INTRODUCTION

EventX is a brand new music event that is to be held in different locations with different artists. The company running this event needs help in designing a new and vibrant website in order to market the event and sell tickets.

DESCRIPTION OF PROJECT AND TASKS

This module involves knowledge about website design, website layout techniques, client side scripting, and server side scripting, all combined in one CMS project. The CMS used for this module is the most common global CMS: WordPress.

DESIGN

Competitors will design a responsive landing page for the music event 'EventX'. The website will be developed as a WordPress child-theme for the Blankslate theme. The child-theme will be supplemented with a number of plugins. The company has prepared content and has requested you to implement the provided layout. Your tasks are to design the logo, style guide and website. The logo and website design needs to be developed for the target audience for EventX: Young people between 18 and 25 years old.

Create a design for a responsive front page for the website so it will be accessible and look beautiful in other devices:

Desktop: 1440px,

Tablet: 768px,

Mobile: 320px.

Add the designs to a mockup to present your website.

You should save the mockup file as: Mockup_EventX.*. The mockup should be saved as *.png or *.psd or *.ai.

The company wants their main page to have at least these elements:

- Image Banner
- About the company (placeholder text)
- Contact Info
- Footer with copyrights and three links to social media
- Tap to open the hidden sidebar

Style Guide

1. Color scheme for
    a. Typography
    b. Buttons
    c. Labels
    d. Background
2. Typography
    a. Main heading
    b. Subpage title
    c. Section header
    d. Buttons
    e. Labels
    f. Paragraphs
3. Grids and spacing, viewports
    a. Mobile
    b. Tablet
    c. Desktop

4. Layout
    a. Navigation
    b. Tables
    c. Lists
    d. Dialog
    e. Form elements
    f. Buttons

Because there will be another developer that will later develop the website too. You need to make a style guide for future development.

The designs should be saved as: Desktop_EventX.*, Tablet_EventX.* and Mobile_EventX.*.

The designs should be saved as *.png and .psd or .ai.

TECHNIQUE

The website layout should be developed with the ability to add menu items and content without damaging the design. In the future some widgets will be placed in the sidebar. Make sure this sidebar can easily be activated or deactivated without damaging the design.

The layout of the website should be identical to your designs, but also needs to scale without damaging the design when scaling the browser window between 320px and 1440px. You should implement the HTML and CSS W3C standards for proper SEO support.

Clicking at post items may not cause a page refresh, but post content will be loaded asynchronous from the server.

The company wants you to use the JQuery image slider they paid for a year ago. You need to rebuild the slider to a proper WordPress plugin.

CONFIGURATION

For safety reasons two user profiles needs to be created:

1. Admin - access to the complete WordPress dashboard.
    a. Username: adMinX
    b. Password: Never4get
2. Contributor - access to the WordPress dashboard except the Appearance, Plugins, Users and Tools.
    a. Username: uSure
    b. Password: Never4get2

3. Editor - Able to accept or decline news created by contributor
    a. Username: eDitor
    b. Password: Never4get3

The Wordpress login page should have the logo of EventX.

The provided plugins should be used to make the website SEO friendly, handle website security and to analyze the visitors of the website.

WEBSITE STRUCTURE

The website consists of several sections described below:

FEATURED SECTION

The Featured section contains a headline, sub headline and an interactive image slider with images of the booked artists. When hovering an image, an mp3 track of the corresponding artist is played. The track is paused when the mouse leaves the image. For the image slider you need to use the delivered JQuery slider. The header section should have an energetic appearance, appropriate to the event. The client wants to have the ability to add, remove or change the images and mp3 files of the slider using the WordPress dashboard.

NEWS SECTION

The News section contains posts about the event. The News section should have a readable and clean design. The client does not have news items; you can use placeholder text to show at least 3 news items. Each news item contains title, date, shortened content and a 'read more' button. By clicking the button the full content will be loaded from the server, without a page refresh (asynchronous).

SAVE THE DATE SECTION

EventX starts at Thursday, 19 October 2019 at 18.00. The Save The Date section contains the date and place of the event and a countdown timer with days, hours, minutes and seconds until the start of the event. The Save The Date section should have an exciting design to engage the target audience. The client wants to have the ability to add, remove or change the date and place of the event using the WordPress dashboard.

BUY YOUR TICKETS SECTION

The Buy Your Tickets section leads to the external website of www.ticketmaster.com This section should be designed attractive and with a clear call to action. The client wants to have the ability to add, remove or change the Internet address linked at the button using the WordPress dashboard.

SIDEBAR SECTION

The sidebar section will be used to show the sponsors of the event and there should be space to add some widgets.

The sidebar should be added to the left side of all pages. The sidebar is hidden but has a small tab to open the sidebar. When user clicks the sidebar, give additional slide or fade in animation to make the entrance of the sidebar appealing.

The client wants to have the ability to add, remove or change logo images and company names for sponsors and website widgets using the WordPress dashboard.

FOOTER SECTION

The footer section contains social media icons and a possibility to leave an e-mail address for subscribing at the newsletter. The client wants to have the ability to add, remove or change social media icons and their corresponding links using the WordPress dashboard.

INSTRUCTIONS TO THE COMPETITOR

1. Save your files in your working directory on the server called "**XX_Module_CMS**".
2. Save your design, style guide and mockup files in this location: **module_cms/design**
3. Save your database dump into a file named "**db-dump.sql**" placed inside the module_cms directory
4. Assessment will be done on the files and data in the database named "**module_cms_db"** on your computer.

MODULE SERVER

CONTENTS

This module has the following files:

- MODULE_SERVER.doc
- MODULE_SERVER_MEDIA.zip

INTRODUCTION

Trilu, task management website, ask you to make their minimum viable product. Your task is to implement the backend with Laravel PHP Framework and frontend with JavaScript Framework (VueJS, AngularJS, Angular, or ReactJS). The front-end design skeleton is provided. The detail description and tools that you can use will be described below.

DESCRIPTION OF PROJECT AND TASKS

API List:

Use provided ERD to make your database. Create dummy users on users table (password is hashed using bcrypt):

username	password	First name	Last name
john.doe	12345	John	Doe
richard.roe	12345	Richard	Roe
jane.poe	12345	Jane	Poe

These are the list of web service endpoint that requested by **Trilu**:

1. **Authentication**

   a. **Register**

   URL: [domain]/v1/auth/register

   Description: For client to register new user

   Method: POST

   Request Parameter:

   - Body:
     - First Name
     - Last Name
     - Username
     - Password

   Validation:

   - First Name must be alphabet only, length between 2 – 20
   - Last name must be alphabet only, length between 2 – 20
   - Username only consist of alphanumeric, underscore '_', or dot '.', length between 5 – 12
   - Username must be unique
   - Password length between 5 - 12

   Response:

   - If register success, registered user automatically logged in:
     Header: response status: 200

Body:

- o Token (authorization token generated by the system from logged in user id with bcrypt hashing method)
- o Role

- If input validation failed:

  Header: response status: 422

  Body: message: invalid field

b. Login

URL: [domain]/v1/auth/login

Description: For client to generate and get login token using username and password. Username and password must be valid.

Method: POST

Request Parameter:

- Body:
  - o Username
  - o Password

Response:

- If login success:

  Header: response status: 200

  Body:

  - o Token (authorization token generated by the system from logged in user id with bcrypt hashing method)
  - o Role

- If login failed (username or password do not match or empty):

  Header: response status: 401

  Body: message: invalid login

c. Logout

URL: [domain]/v1/auth/logout?token={*AUTHORIZATION_TOKEN*}

Description: For server to make token invalid

Method: GET

Response:

- If logout success:

  Header: response status: 200

  Body:

  - o message: logout success

- If logout failed (token invalid):

  Header: response status: 401

  Body: message: unauthorized user

2. **Board**

   a. Create new board

   URL: [domain]/v1/board?token={*AUTHORIZATION_TOKEN*}

Description: For client to create new board. Assign creator as team member when board created.

Method: POST

Request Parameter:

- Body:
    - o Name

Validation:

- name must be filled

Response:

- If success:

    Header: response status: 200

    Body: message: create board success

- If input validation failed:

    Header: response status: 422

    Body: message: invalid field

- If unauthorized user access it (only logged in user can access this endpoint):

    Header: response status: 401

    Body: message: unauthorized user

b. Update board

URL: [domain]/v1/board/{*board_id*}?token={*AUTHORIZATION_TOKEN*}

Description: For client to update existing board by id

Method: PUT

Request Parameter:

- Body:
    - o Name

Validation:

- name must be filled

Response:

- If success:

    Header: response status: 200

    Body: message: update board success

- If input validation failed:

    Header: response status: 422

    Body: message: invalid field

- If unauthorized user access it (only team member can access this endpoint):

    Header: response status: 401

    Body: message: unauthorized user

c. Delete board

URL: [domain]/v1/board/{*board_id*}?token={*AUTHORIZATION_TOKEN*}

Description: For client to delete existing board by id

Method: DELETE

Response:

- If success:

  Header: response status: 200

  Body: message: delete board success

- If unauthorized user access it (only creator can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

d. Get all boards

URL: [domain]/v1/board?token={*AUTHORIZATION_TOKEN*}

Description: For client to get all boards data based on logged in user, as a member or creator.

Method: GET

Response:

- If success:

  Header: response status: 200

  Body: all board data in json (consists of id, name, creator_id)

- If unauthorized user access it (only logged in user can access this endpoint):

  Header: response status: 401

e. Open board

URL: [domain]/v1/board/{*board_id*}?token={*AUTHORIZATION_TOKEN*}

Description: For client to get board detail based on board_id. Lists and cards sorted by column order.

Method: GET

Response:

- If success:

  Header: response status: 200

  Body: eager load all board data in json (consists of all team members, all lists of board and cards inside list)

Format:

```
{
  "id": [board_id],
  "name": [board_name],
   creator_id: [creator_id],
  "members": [
    {
      "id": [user_id],
      "first_name": [first name],
      "last_name": [last name],
      "initial": [generated from first name and last name]
    },
    …
  ],
  "lists": [
    {
      "id": [list_id],
      "name": [list_name],
      "order": [order],
      "cards": [
        {
          "card_id": [card_id],
          "task": [card_task],
          "order", [order]
        },
        …
      ]
    },
    …
  ]
}
```

- If unauthorized user access it (only team member can access this endpoint):
  Header: response status: 401

f. Add team member

URL: [domain]/v1/board/*{board_id}*/member?token={*AUTHORIZATION_TOKEN*}

Description: For client to add team member to the board.

Method: POST

Request Parameter:

- Body:
  - o Username

Validation:

- username must be existing in user table

Response:

- If success:

  Header: response status: 200

  Body: message: add member success

- If input validation failed:

  Header: response status: 422

  Body: message: user did not exist

- If unauthorized user access it (only team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

g. Remove team member

URL:

[domain]/v1/board/*{board_id}*/member/*{user_id}*?token={*AUTHORIZATION_TOKEN*}

Description: For client to delete team member from the board using user id.

Method: DELETE

Response:

- If success:

  Header: response status: 200

  Body: me   ssage: remove member success

- If unauthorized user access it (only team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

3. List

a. Create new list

URL: [domain]/v1/board/*{board_id}*/list?token={*AUTHORIZATION_TOKEN*}

Description: For client to create new list to the board

Method: POST

Request Parameter:

- Body:
  - Name

Validation:

- name must be filled

Response:

- If success:

  Header: response status: 200

  Body: message: create list success

- If input validation failed:

  Header: response status: 422

Body: message: invalid field

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

b. Update list

URL: [domain]/v1/board/{*board_id*}/list/*{list_id}*?token={*AUTHORIZATION_TOKEN*}

Description: For client to update existing list by id

Method: PUT

Request Parameter:

- Body:
  - Name

Validation:

- name must be filled

Response:

- If success:

  Header: response status: 200

  Body: message: update list success

- If input validation failed:

  Header: response status: 422

  Body: message: invalid field

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

c. Delete list

URL: [domain]/v1/board/{*board_id*}/list/*{list_id}*?token={*AUTHORIZATION_TOKEN*}

Description: For client to delete existing list by id

Method: DELETE

Response:

- If success:

  Header: response status: 200

  Body: message: delete list success

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

d. Move list to right

URL:
[domain]/v1/board/{*board_id*}/list/*{list_id}*/right?token={*AUTHORIZATION_TOKEN*}

Description: For client to switch the order of the selected list with the list on the right

Method: POST

Response:

- If success:

  Header: response status: 200

  Body: message: move success

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

e. Move list to left

URL:
[domain]/v1/board/{*board_id*}/list/{*list_id*}/left?token={*AUTHORIZATION_TOKEN*}

Description: For client to switch the order of the selected list with the list on the left

Method: POST

Response:

- If success:

  Header: response status: 200

  Body: message: move success

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

4. Card

a. Create new card

URL:
[domain]/v1/board/{*board_id*}/list/{*list_id*}/card?token={*AUTHORIZATION_TOKEN*}

Description: For client to create new card to the list

Method: POST

Request Parameter:

- Body:
  - Task

Validation:

- Task must be filled

Response:

- If success:

  Header: response status: 200

  Body: message: create card success

- If input validation failed:

Header: response status: 422

Body: message: invalid field

- If unauthorized user access it (only board team member can access this endpoint):

    Header: response status: 401

    Body: message: unauthorized user

b. Update card

URL:
[domain]/v1/board/{*board_id*}/list/{*list_id*}/card/{*card_id*}?token={*AUTHORIZATION_TOKEN*}

Description: For client to update existing card by id

Method: PUT

Request Parameter:

- Body:
    - Task

Validation:

- Task must be filled

Response:

- If success:

    Header: response status: 200

    Body: message: update card success

- If input validation failed:

    Header: response status: 422

    Body: message: invalid field

- If unauthorized user access it (only board team member can access this endpoint):

    Header: response status: 401

    Body: message: unauthorized user

c. Delete card

URL:
[domain]/v1/board/{*board_id*}/list/{*list_id*}/card/{*card_id*}?token={*AUTHORIZATION_TOKEN*}

Description: For client to delete existing card by id

Method: DELETE

Response:

- If success:

    Header: response status: 200

    Body: message: delete card success

- If unauthorized user access it (only board team member can access this endpoint):

    Header: response status: 401

    Body: message: unauthorized user

d. Move up card

URL: [domain]/v1/card/*{card_id}*/up?token={*AUTHORIZATION_TOKEN*}

Description: For client to switch the order of the selected card with the card above

Method: POST

Response:

- If success:

  Header: response status: 200

  Body: message: move success

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

e. Move down card

URL: [domain]/v1/card/*{card_id}*/down?token={*AUTHORIZATION_TOKEN*}

Description: For client to switch the order of the selected card with the card below

Method: POST

Response:

- If success:

  Header: response status: 200

  Body: message: move success

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

f. Move card to another list

URL: [domain]/v1/card/*{card_id}*/move/*{list_id}*?token={*AUTHORIZATION_TOKEN*}

Description: For client to move card to another list. List must be in the same board.

Method: POST

Response:

- If success:

  Header: response status: 200

  Body: message: move success

- If list id is not in the same board:

  Header: response status: 422

  Body: message: move list invalid

- If unauthorized user access it (only board team member can access this endpoint):

  Header: response status: 401

  Body: message: unauthorized user

Make sure that all validations are done in **both backend and frontend**. The complete minimum viable product for **Trilu** system should cover the following requirement:

Menu	Detail
Login & Register	- User can register into the system on the register page - User can login (and logout) into the system on the start page of the application - After login, user directed to "home" menu - Links to the Home and Logout are always visible on the top part of the page while the user is logged in
Home	- On "home" menu, user can add new board or access board if the user is the team member - User can update board's name by clicking the board name, type new name, then press Enter to submit - Creator of the board can delete a board by deleting its entire name, then press Enter to submit - Make sure your system is preventing users to view and access unauthorized board
Board	- On "board" menu, user can manage board: members, lists, and cards **Member** - User can add new member using username - User can remove member by clicking its initial then click confirmation button **List** - User can add new list to the board - User can update list's name by clicking the list name, type new name, then press Enter to submit - User can delete a list by deleting its entire name, then press Enter to submit - User can move list left and right **Card** - User can add new card to a list - User can update card's task by clicking the card, type new task, then press Enter to submit - User can delete a card by deleting its entire task, then press Enter to submit - User can move card up and down in the list - User can move card to another list by click the card, then click another list.

ERD

You can use and improve ERD below: