

SKRIPSI

**PEMBANGUNAN PERANGKAT LUNAK BERBASIS WEB
DENGAN MEMANFAATKAN API BYBIT**



Jason

NPM: 6181801061

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2023**

UNDERGRADUATE THESIS

**BUILDING A WEB-BASED SOFTWARE UTILIZING BYBIT
API**



Jason

NPM: 6181801061

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2023**

LEMBAR PENGESAHAN

PEMBANGUNAN PERANGKAT LUNAK BERBASIS WEB DENGAN MEMANFAATKAN API BYBIT

Jason

NPM: 6181801061

Bandung, 19 Januari 2023

Menyetujui,

Pembimbing

Digitally signed
by Raymond
Chandra Putra

Raymond Chandra Putra, M.T.

Ketua Tim Penguji

Digitally signed
by Pascal

Alfadian Nugroho

Pascal Alfadian, Nugroho, M.Comp.

Anggota Tim Penguji

Digitally signed
by Keenan

Adiwijaya Leman

Keenan Adiwijaya Leman, M.T.

Mengetahui,

Ketua Program Studi

Digitally signed
by Mariskha Tri
Adithia

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PEMBANGUNAN PERANGKAT LUNAK BERBASIS WEB DENGAN MEMANFAATKAN API BYBIT

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 19 Januari 2023



Jason
NPM: 6181801061

ABSTRAK

Perdagangan mata uang kripto merupakan suatu tindakan perdagangan jual atau beli yang dilakukan pelaku pasar aset digital untuk memperoleh keuntungan. Kegiatan jual atau beli *cryptocurrency* dapat dilakukan pada bursa yang menyediakan jasa dan likuiditas untuk memperdagangkan aset tersebut. Bybit merupakan bursa yang menyediakan layanan bertransaksi serta berdagang mata uang kripto. Salah satu kekurangan dari Bybit adalah tidak menyediakan data performa perdagangan secara lengkap, sehingga pelaku pasar sulit untuk mengevaluasi dan melacak performa mereka di pasar. Meski demikian, Bybit menyediakan layanan API untuk mengambil data seperti posisi perdagangan, info saldo, dan sejarah perdagangan.

Pada skripsi ini, telah dirancang sebuah perangkat lunak berbasis web memanfaatkan API Bybit . Perangkat lunak berbasis web yang dibangun memiliki fitur yaitu melihat info saldo, sejarah perdagangan, persentase kemenangan, rasio jual dan beli, keuntungan dan kerugian berdasarkan hari, keuntungan dan kerugian terkecil, keuntungan dan kerugian terbesar, persentase kemenangan kontrak beli, persentase kemenangan kontrak jual, kontrak perdagangan, dan membuka dan menutup kontrak perdagangan. Antarmuka perangkat lunak berbasis web ini dibangun menggunakan *framework* Vue.js dan Tailwind. *Server-side* pada perangkat lunak berbasis web ini dibangun menggunakan Express.js.

Pembangunan perangkat lunak *cryptocurrency* berbasis web memanfaatkan API Bybit telah berhasil dibangun dan dilakukan pengujian. Berdasarkan pengujian eksperimental dan pengujian fungsional yang dilakukan, perangkat lunak berbasis web ini berjalan dengan baik dan semua fitur sudah berhasil dicoba oleh penguji. Perangkat lunak berbasis web yang dibangun sudah dapat membantu menyelesaikan masalah evaluasi serta pelacakan performa pelaku pasar.

Kata-kata kunci: *cryptocurrency*, Bybit, API, Tailwind, Express.js, Node.js

ABSTRACT

Cryptocurrency trading is an act of buying or selling digital assets that are done by traders to gain profits. The activity of buying and selling cryptocurrency can be done on an exchange that provides the service and liquidity to trade the corresponding asset. Bybit is an exchange that offers transaction services and trading cryptocurrency. One of the drawbacks of Bybit is not providing enough data to evaluate performance, and it is hard for traders to evaluate and track their performance in the market because of it. Despite that, Bybit offers an API service to fetch data such as contract position, wallet information, and trading history.

In this thesis, a web-based software utilizing Bybit API is designed. The Web-based software is built with features such as viewing wallet information, trading history, win rate, buy and sell ratio, profit and loss grouped by day, smallest profit and loss, biggest profit and loss, buy contract win rate, sell contract win rate, trading contract, and opening and closing a trading contract. The interface of this web-based software is built using frameworks such as Vue.js and Tailwind. The server-side of the software is built using frameworks such as Express.js.

Building a web-based cryptocurrency software utilizing Bybit API is successfully built and tests are carried out. Based on the experimental and functional testing that is done, the web-based software runs well and all the features are successfully tried by the testers. The web-based software that is built can contribute to solving problems such as evaluating and also tracking the performance of traders.

Keywords: Cryptocurrency, Bybit, API, Tailwind, Express.js, Node.js

Saya persembahkan skripsi ini untuk orang tua, teman-teman, dan dosen yang telah membantu saya sampai pada titik ini.

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa karena dengan rahmat dan karunia-Nya, skripsi dengan judul Pembangunan Perangkat Lunak Berbasis Web Dengan Memanfaatkan API Bybit dapat diselesaikan dengan baik dan tepat waktu. Pada saat proses pembuatan skripsi penulis menghadapi berbagai hambatan dan rintangan, namun hambatan dan rintangan tersebut tidak menghentikan penulis untuk menyelesaikan skripsi ini. Karena itu, penulis mengucapkan terima kasih sebesar-besarnya kepada:

- Untuk kedua orang tua dan saudara yang telah memberikan semangat selama pengerjaan skripsi.
- Bapak Raymond Chandra Putra, sebagai dosen pembimbing yang selalu memberikan bimbingan serta arahan agar skripsi ini dapat terealisasi dan tujuan dari penulisan skripsi ini dapat tercapai.
- Teman-teman dari *channel Discord* server kunci (Clement, Michael, Nicholas, Julyus, Bernadus, Ivan, dan Christopher) karena telah memberi semangat setiap penulis mengeluh dalam mengerjakan skripsi.
- Gabriel yang berkala menanyakan status pengerjaan skripsi ini yang secara tidak langsung memberikan semangat kepada penulis untuk menyelesaikan skripsi ini.
- Dan untuk pihak-pihak lain yang telah membantu serta mendukung namun namanya belum disebutkan.

Bandung, Januari 2023

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
DAFTAR KODE PROGRAM	xxiv
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Metodologi	3
1.6 Sistematika Pembahasan	4
DAFTAR NOTASI	1
2 LANDASAN TEORI	5
2.1 Cryptocurrency	5
2.1.1 Blockchain	5
2.1.2 Jenis-jenis Cryptocurrency	6
2.1.3 Bursa Cryptocurrency	6
2.2 Bybit	6
2.2.1 Fitur Bybit	6
2.2.2 API Bybit	8
2.2.3 Endpoint	9
2.3 Pemrograman Berbasis Web	14
2.3.1 Front-End & Back-end	14
2.3.2 Web Service	15
2.3.3 Web Framework	16
2.4 Express	16
2.5 JSON Web Token	19
2.6 Vue.js	20
2.7 Tailwind	23
2.8 MySQL	25
2.9 Node.js	26
2.10 chartJS	27
3 ANALISIS	29
3.1 Analisis Masalah	29
3.2 Analisis API Bybit	31

3.2.1	Analisis Fitur API Bybit	31
3.2.2	Analisis Kebutuhan API Bybit	32
3.3	Analisis Pemilihan Tech Stack	33
3.4	Analisis Kebutuhan Perangkat Lunak	34
3.4.1	Analisis Pengguna	34
3.4.2	Spesifikasi Perangkat Lunak	34
3.5	Analisis Arsitektur	35
3.5.1	Diagram Client-Server	35
3.5.2	Pembagian Modul Pengerjaan	36
3.6	Analisis Perangkat Lunak	38
3.6.1	Use Case Diagram	39
3.6.2	Use Case Scenario	39
3.6.3	Swimlane Diagram	43
3.6.4	Data Flow Diagram	49
3.6.5	<i>Entity Relationship Diagram</i>	51
4	PERANCANGAN	53
4.1	Perancangan Basis Data	53
4.1.1	Perancangan Relasional Basis Data	53
4.1.2	Perancangan Fisik Basis Data	53
4.2	<i>Data Flow Diagram Level 2</i>	54
4.2.1	<i>Data Flow Diagram Level 2 user</i>	54
4.2.2	<i>Data Flow Diagram Level 2 api_info</i>	55
4.2.3	<i>Data Flow Diagram Level 2 token</i>	57
4.2.4	<i>Data Flow Diagram Level 2 API Bybit</i>	58
4.3	Perancangan Antarmuka	59
4.3.1	Screenflow	59
4.3.2	Perancangan <i>Wireframe</i> Antarmuka	61
4.4	Perancangan Komponen Vue	80
4.5	Perancangan Endpoint	90
4.5.1	Perancangan API <i>user</i>	90
4.5.2	Perancangan API <i>api_info</i>	91
4.5.3	Perancangan API <i>JWT</i>	92
4.5.4	Perancangan API Bybit	93
5	IMPLEMENTASI DAN PENGUJIAN	95
5.1	Implementasi	95
5.1.1	Lingkungan Implementasi	95
5.1.2	Implementasi Basis Data	95
5.1.3	Implementasi Antarmuka	96
5.2	Pengujian	105
5.2.1	Pengujian Fungsional	105
5.2.2	Pengujian Eksperimental	112
5.2.3	Analisis Hasil Pengujian	115
6	KESIMPULAN DAN SARAN	117
6.1	Kesimpulan	117
6.2	Saran	117
	DAFTAR REFERENSI	119
	A KODE PROGRAM	121

DAFTAR GAMBAR

2.1	Contoh <i>blockchain</i> pada Bitcoin, salah satu mata uang kripto	5
3.1	Tampilan halaman <i>API Management</i> pada Bybit	32
3.2	Tampilan halaman <i>API Management</i> setelah menekan tombol <i>Create New Key</i> pada Bybit	32
3.3	Tampilan halaman <i>API Management</i> setelah memasukkan kode <i>Google 2FA</i>	33
3.4	Diagram <i>client-server</i>	35
3.5	Struktur modul perangkat lunak berbasis web	36
3.6	Use-case diagram	39
3.7	<i>Swimlane diagram</i> untuk halaman <i>home</i> pada admin	43
3.8	<i>Swimlane diagram</i> untuk menyetujui dan menghapus data <i>user</i>	43
3.9	<i>Swimlane diagram</i> untuk menghapus data <i>user</i> yang telah disetujui	44
3.10	<i>Swimlane diagram</i> untuk melihat dan memperbaiki data salah satu <i>user</i>	44
3.11	<i>Swimlane diagram</i> untuk menambah <i>api_info</i>	45
3.12	<i>Swimlane diagram</i> untuk memperbaiki dan menghapus data <i>api_info</i>	45
3.13	<i>Swimlane diagram</i> untuk enkripsi data <i>api_info</i>	46
3.14	<i>Swimlane diagram</i> untuk melihat info saldo	47
3.15	<i>Swimlane diagram</i> untuk melihat sejarah perdagangan	47
3.16	<i>Swimlane diagram</i> untuk melihat analitik perdagangan	48
3.17	<i>Swimlane diagram</i> untuk melihat dan menutup kontrak perdagangan	48
3.18	<i>Swimlane diagram</i> untuk membuka kontrak perdagangan	49
3.19	DFD Level 0	49
3.20	DFD Level 1	50
3.21	Gambar <i>Entity Relationship Diagram</i>	51
4.1	Gambar perancangan relational basis data	53
4.2	DFD Level 2 Mengelola data <i>user</i>	54
4.3	DFD Level 2 Mengelola data <i>api_info</i>	56
4.4	DFD Level 2 Mengelola token <i>user</i> , <i>admin</i> dan <i>api_info</i>	57
4.5	DFD Level 2 Mengelola API Bybit	58
4.6	<i>Screenflow</i> untuk antarmuka perangkat lunak.	60
4.7	Halaman <i>register</i>	61
4.8	Halaman <i>login</i>	63
4.9	Halaman <i>home</i> admin	64
4.10	Halaman <i>approved users</i>	65
4.11	Halaman <i>pending approvals</i>	67
4.12	Halaman <i>personal info</i>	68
4.13	Halaman <i>personal info</i> setelah menekan <i>icon edit</i>	68
4.14	Halaman <i>add API key</i>	70
4.15	Halaman <i>API info</i>	71
4.16	Halaman <i>API info</i> setelah menekan <i>edit icon</i> pada salah satu data	72
4.17	Halaman <i>wallet balances</i>	74
4.18	Halaman <i>trading history</i> sebelum menekan tombol <i>submit</i>	75

4.19	Halaman <i>trading history</i> setelah menekan tombol <i>submit</i>	76
4.20	Halaman <i>trading analytics</i>	77
4.21	Halaman <i>trading hub</i>	79
5.1	Halaman <i>register</i>	97
5.2	Halaman <i>login</i>	97
5.3	Halaman <i>home admin</i>	98
5.4	Halaman <i>approved users</i>	98
5.5	Halaman <i>pending approvals</i>	99
5.6	Halaman <i>personal info</i>	99
5.7	Halaman <i>personal info</i> setelah menekan <i>icon edit</i>	100
5.8	Halaman <i>add API key</i>	100
5.9	Halaman <i>API info</i>	101
5.10	Halaman <i>API info</i> setelah menekan <i>edit icon</i> pada salah satu data	101
5.11	Halaman <i>wallet balances</i>	102
5.12	Halaman <i>trading history</i> sebelum menekan tombol <i>submit</i>	102
5.13	Halaman <i>trading history</i> setelah menekan tombol <i>submit</i>	103
5.14	Halaman <i>trading analytics</i>	103
5.15	Halaman <i>trading analytics</i>	104
5.16	Halaman <i>trading hub</i>	104
5.17	Pertanyaan pertama pada kuesioner	112
5.18	Pertanyaan kedua pada kuesioner	113
5.19	Pertanyaan ketiga pada kuesioner	113
5.20	Pertanyaan keempat pada kuesioner	114
5.21	Pertanyaan kelima pada kuesioner	114
5.22	Pertanyaan keenam pada kuesioner	114
5.23	Pertanyaan ketujuh pada kuesioner	115

DAFTAR TABEL

2.1	Tabel parameter pada <i>endpoint</i> untuk mendapatkan sejarah perdagangan	9
2.2	Tabel respons untuk mendapatkan sejarah perdagangan	9
2.3	Tabel parameter pada <i>endpoint</i> untuk mendapatkan nilai saldo	10
2.4	Tabel respons untuk mendapatkan nilai saldo	10
2.5	Tabel respons untuk mendapatkan waktu UTC	11
2.6	Tabel parameter pada <i>endpoint</i> untuk mendapatkan informasi terkait pair tertentu	11
2.7	Tabel respons untuk mendapatkan informasi <i>pair</i> tertentu	11
2.8	Tabel parameter pada <i>endpoint</i> untuk mendapatkan informasi terkait kontrak yang dibuka oleh user	12
2.9	Tabel respons untuk mendapatkan informasi terkait kontrak yang dibuka oleh user	12
2.10	Tabel parameter pada <i>endpoint</i> untuk membuka dan menutup sebuah kontrak	13
2.11	Tabel respons untuk membuka dan menutup sebuah kontrak	14
2.12	Tabel penjelasan <i>method</i> untuk <i>request</i> dan <i>response</i>	18
3.1	Tabel penjelasan modul pengerjaan perangkat lunak	37
3.2	Deskripsi atribut pada entitas admin	52
3.3	Deskripsi atribut pada entitas user	52
3.4	Deskripsi atribut pada entitas <i>api_info</i>	52
4.1	Perancangan fisik tabel <i>user</i> , <i>api_info</i> ,	54
5.1	Pengujian halaman <i>login</i>	105
5.2	Pengujian halaman <i>register</i>	105
5.3	Pengujian halaman utama admin	106
5.4	Pengujian halaman <i>approval users</i>	106
5.5	Pengujian halaman <i>pending approval</i>	106
5.6	Pengujian halaman <i>personal info</i>	107
5.7	Pengujian halaman <i>add API keys</i>	107
5.8	Pengujian halaman <i>api info</i>	108
5.9	Pengujian halaman <i>wallet balance</i>	109
5.10	Pengujian halaman <i>trading history</i>	109
5.11	Pengujian halaman <i>trading analytics</i>	110
5.12	Pengujian halaman <i>trading hub</i>	111
5.13	Tabel demografik responden	112

DAFTAR KODE PROGRAM

2.1	Contoh penggunaan <i>routing</i> pada Express.js	17
2.2	Contoh penggunaan <i>application-level middleware</i> pada Express.js	17
2.3	Contoh penggunaan <i>application-level middleware</i> pada Express.js	17
2.4	Contoh penggunaan <i>routing</i> pada Express.js	17
2.5	Contoh penggunaan <i>error-handling middleware</i> pada Express.js	18
2.6	Contoh penggunaan <i>third-party middleware</i> pada Express.js	18
2.7	Contoh objek JSON pada bagian <i>header</i>	19
2.8	Contoh objek JSON pada bagian <i>payload</i>	19
2.9	Contoh sederhana kode pada sebuah file <i>.vue</i>	21
2.10	Contoh kode Vue yang memanfaatkan <i>declarative rendering</i>	21
2.11	Contoh kode Vue yang memanfaatkan <i>reactivity</i>	21
2.12	Contoh kode Vue yang memanfaatkan <i>directives</i>	22
2.13	Contoh potongan kode untuk mengubah <i>font</i> pada suatu <i>text</i>	23
2.14	Contoh potongan kode untuk mengubah <i>border</i> pada suatu elemen	23
2.15	Contoh potongan kode untuk mengubah <i>shadow</i> pada suatu elemen	24
2.16	Contoh potongan kode untuk mengubah <i>margin</i> dan <i>padding</i> pada suatu elemen	24
2.17	Contoh potongan kode untuk mengubah lebar dan tinggi pada suatu elemen	24
2.18	Contoh potongan kode untuk melakukan transformasi pada suatu elemen	25
2.19	Contoh potongan kode untuk mengubah <i>pseudoclass</i> pada suatu elemen	25
2.20	Contoh potongan kode untuk mengubah sistem layout pada suatu elemen	25
5.1	Kode untuk membuat tabel admin	95
5.2	Kode untuk membuat tabel user	96
5.3	Kode untuk membuat tabel <i>apoi_info</i>	96
A.1	<i>db-configuration.js</i>	121
A.2	<i>db.js</i>	121
A.3	<i>webapi-configuration.js</i>	121
A.4	<i>login.js</i>	121
A.5	<i>registration.js</i>	122
A.6	<i>login.js</i>	122
A.7	<i>user-list.js</i>	123
A.8	<i>user-info.js</i>	123
A.9	<i>update-user.js</i>	124
A.10	<i>update-userapproval.js</i>	124
A.11	<i>delete-user.js</i>	125
A.12	<i>api-list.js</i>	125
A.13	<i>api-info.js</i>	126
A.14	<i>add-api.js</i>	126
A.15	<i>update-api.js</i>	127
A.16	<i>delete-api.js</i>	128
A.17	<i>api-token.js</i>	128

A.18	coin-tickers.js	129
A.19	wallet-balance.js	129
A.20	closed-pnl.js	130
A.21	position-list.js	131
A.22	create-position.js	132
A.23	app.js	133
A.24	.HomeAdminView.vue	134
A.25	ApprovedAdminView.vue	136
A.26	PendingAdminView.vue	139
A.27	RegisterUserView.vue	142
A.28	LoginView.vue	143
A.29	PersonalInfoView.vue	144
A.30	AddApiView.vue	148
A.31	ApiInfoView.vue	151
A.32	WalletBalanceView.vue	156
A.33	TradingHistoryView.vue	160
A.34	TradingAnalyticsView.vue	167
A.35	TradingHubView.vue	174
A.36	index.js	179
A.37	main.js	180

BAB 1

PENDAHULUAN

Pada bab ini, akan dijelaskan gambaran umum dari permasalahan yang akan dibahas. Bab ini terdiri dari latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan.

1.1 Latar Belakang

Mata uang kripto adalah sistem alat pembayaran virtual yang berfungsi sebagai media transaksi finansial. Mata uang kripto merupakan salah satu aset digital, sehingga pemilik aset tersebut mempunyai hak untuk menggunakan aset tersebut untuk alat tukar, pembayaran, penyimpanan nilai, investasi, perdagangan, serta penarikan dana yang digunakan untuk pendanaan suatu kegiatan [1]. Di negara tertentu termasuk Indonesia, mata uang kripto merupakan salah satu jenis komoditas, sehingga dapat diperdagangkan dan ditukarkan dengan mata uang suatu negara.

Perdagangan mata uang kripto merupakan sebuah tindakan perdagangan atau jual-beli yang dilakukan pelaku pasar aset digital untuk memperoleh keuntungan. Pelaku pasar memperoleh keuntungan dan kerugian dari perbedaan harga jual dan harga beli mata uang kripto tersebut terhadap mata uang tertentu. Kegiatan jual-beli pada mata uang kripto dapat dilakukan pada suatu bursa yang menyediakan jasa dan likuiditas untuk memperdagangkan *aset* tersebut.

Bybit merupakan bursa yang menyediakan layanan untuk bertransaksi serta memperdagangkan mata uang kripto. Bybit merupakan suatu bursa yang menyediakan ratusan aset mata uang kripto untuk diperdagangkan dan merupakan salah satu bursa dengan likuiditas terbesar di dunia. Bybit merupakan salah satu bursa *cryptocurrency* paling populer untuk para pelaku pasar, dengan volume perdagangan perhari mencapai 2 miliar USD.

Bybit menyediakan dua mode eksekusi pembukaan dan penutupan kontrak, yaitu *one-way mode* dan *hedge mode*. *One-way mode* merupakan jenis pemesanan kontrak yang mengizinkan pelaku pasar untuk membuka sebuah posisi hanya pada arah yang spesifik, yaitu *buy contract* atau *sell contract*. Jika pelaku pasar memiliki *buy contract* yang sedang berjalan, kemudian dilakukan *sell contract*, akan terjadi penutupan kontrak yang berlawanan arah yaitu *buy*, kemudian sebaliknya. Mode eksekusi *one-way mode* biasanya digunakan oleh entitas dengan resiko yang tinggi, seperti *retail trader*. *Hedge mode* merupakan jenis pemesanan kontrak yang mengizinkan pelaku pasar untuk membuka posisi pada dua arah yang berbeda. *Hedge mode* digunakan untuk meminimalisir eksposur pada suatu arah, dan biasanya digunakan oleh entitas dengan resiko yang rendah seperti *hedge fund* dan *trading firm*.

Bybit menyediakan dua jenis *order* untuk pembukaan dan penutupan kontrak, yaitu *market order* dan *limit order*. *Market order* merupakan jenis pemesanan kontrak jual dan beli pada harga pasar saat ini yang diseksekusi secara langsung. *Limit order* merupakan jenis pemesanan kontrak jual dan beli pada harga tertentu yang akan diseksekusi jika harga saat ini menyentuh harga *limit*

yang dipasang oleh pelaku pasar.

Salah satu kekurangan bursa Bybit adalah tidak menyediakan data performa perdagangan secara lengkap, sehingga sulit bagi pengguna untuk mengevaluasi dan melacak performa mereka di pasar. Data seperti persentase kemenangan (*win rate*) dan total keuntungan/kerugian pada rentang waktu tertentu tidak disediakan oleh bursa Bybit. Persentase kemenangan dan total keuntungan/kerugian pada rentang waktu tertentu merupakan metrik penting bagi para pelaku pasar untuk mengevaluasi performa perdagangan. Bursa *cryptocurrency* lain seperti Binance menyediakan metrik/data tersebut sehingga pelaku pasar yang memakai bursa Binance lebih mudah untuk melakukan evaluasi terhadap performa perdagangan mereka.

Meski demikian, Bybit menyediakan API untuk mengambil data dari sejarah perdagangan yang telah dieksekusi oleh pengguna. Untuk memudahkan pengguna dalam melakukan evaluasi performa terhadap pasar mata uang kripto, dibangun sebuah perangkat lunak berbasis web dengan memanfaatkan API Bybit. Fitur-fitur yang dapat dibuat pada perangkat lunak dengan memanfaatkan API Bybit yaitu melihat saldo, sejarah perdagangan, melihat posisi dari kontrak yang masih berjalan, persentase kemenangan, rasio jual dan beli, total keuntungan dan kerugian, beserta data yang lebih rinci seperti penjualan serta pembelian terbaik, keuntungan dan kerugian berdasarkan hari, keuntungan dan kerugian dari satu sisi baik pembelian maupun penjualan. Fitur tambahan seperti membuka dan menutup kontrak/posisi juga akan ditambahkan untuk menambah fungsionalitas dari perangkat lunak yang akan dibangun. Pembangunan perangkat lunak berbasis web dengan fitur-fitur tersebut akan membantu evaluasi performa pada pasar *cryptocurrency* karena kurangnya data sejarah perdagangan dari bursa Bybit.

Untuk membantu melakukan *styling* pada halaman perangkat lunak tanpa diperlukannya pembuatan *custom* CSS, digunakan Tailwind.css. Tailwind merupakan *framework* CSS yang digunakan untuk membangun desain antarmuka yang didasari dari sekumpulan *utility class*. *Utility class* tersebut dapat dilakukan kustomisasi sesuai kebutuhan desain antarmuka. Beberapa *utility class* yang dapat dikustomisasi seperti *typography*, *border*, *margin*, *padding*, *shadow*, *width*, dan *height* akan memberikan kemudahan *styling* dalam pengembangan perangkat lunak. Untuk memudahkan *styling* dengan *utility class* yang disediakan Tailwind, dibutuhkan sebuah *framework* untuk bagian *front-end* yang mendukung antarmuka yang dinamis serta interaktif.

Vue.js merupakan sebuah *front-end framework* untuk Javascript yang dibangun diatas HTML, CSS, dan Javascript. Fitur utama yang terdapat pada Vue.js yaitu penggunaan *single-file component*, *declarative rendering*, *reactivity* dan *directives*. *Tools* yang tersedia tersebut dapat digunakan oleh pengembang untuk membuat antarmuka yang efisien, dinamis, dan interaktif. Agar perangkat lunak yang dibangun dapat terhubung serta menampilkan data yang didapat dari API Bybit, dibutuhkan sebuah *back-end framework* untuk menjalin hubungan antara antarmuka dengan server Bybit.

Agar pemanggilan API eksternal ke *server* Bybit dapat terjalin dengan antarmuka, digunakan Express.js. Express.js merupakan *back-end framework* yang dibangun diatas Node.js yang digunakan untuk membangun API. Express.js menyediakan fitur seperti *templating engine*, *routing*, dan *middleware* yang dapat mendukung pembuatan aplikasi web dan API sehingga pengembangannya dapat dilakukan dengan lebih cepat dan lebih mudah.

Untuk menambah lapisan keamanan serta melakukan pertukaran informasi, digunakan JSON Web Token. JSON Web Token merupakan sebuah standar keamanan berformat *string* yang berfokus pada otorisasi dan pertukaran informasi antar dua pihak dalam notasi JSON. *String* pada JWT terbagi menjadi tiga bagian yang dipisahkan oleh titik (*.*), yaitu *header*, *payload*, dan *signature*. *Header* merupakan komponen/ bagian dari JWT yang terdiri dari jenis token serta algoritma yang digunakan untuk melakukan operasi *sign*. *Payload* merupakan bagian kedua dari JWT yang terdiri dari pernyataan atau klaim berupa data seperti nama, umur, dan lain-lain. *Signature* merupakan bagian terakhir dari JWT yang terdiri dari hasil enkripsi antara *header* dan *payload* serta sebuah *string* sebagai parameter untuk memverifikasi keaslian dari sebuah JWT.

Pada skripsi ini, dibangun sebuah perangkat lunak *cryptocurrency* berbasis web dengan memanfaatkan API Bybit menggunakan utilitas dari Vue.js, Express.js, dan Tailwind. Aplikasi ini akan memanggil API dari Bybit dan mendapatkan data seputar sejarah perdagangan pada pasar *futures* dan mengkomunikasikan data yang dibutuhkan oleh pengguna untuk keperluan evaluasi performa mereka terhadap pasar.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan di atas, maka dihasilkan beberapa poin yang menjadi masalah. Rumusan masalah pada skripsi ini adalah sebagai berikut:

1. Bagaimana cara kerja dari API bursa Bybit?
2. Bagaimana pembangunan perangkat lunak berbasis web dari pemanggilan API Bybit hingga menjadi fitur-fitur pemanggilan data menggunakan Tailwind, Express.js, dan Vue.js?

1.3 Tujuan

Tujuan yang ingin dicapai oleh skripsi ini adalah untuk menjawab rumusan masalah yang telah dibuat, yaitu :

1. Mengambil data dari API bursa Bybit.
2. Menganalisis, merancang, mengimplementasikan, dan menguji perangkat lunak berbasis web untuk mengkomunikasikan data yang didapatkan dari API bursa Bybit menggunakan Tailwind, Express.js.

1.4 Batasan Masalah

Batasan masalah dari skripsi ini adalah sebagai berikut.

1. Pada perdagangan *cryptocurrency*, terdapat dua *mode* eksekusi pembukaan dan penutupan kontrak, yaitu *one-way mode* dan *hedge mode*. Perangkat lunak hanya bisa menangani *hedge mode* agar meminimalisir resiko dan eksposur pada suatu kontrak.
2. Pada perdagangan *cryptocurrency*, terdapat dua jenis *order*, yaitu *market order* dan *limit order*. Perangkat lunak hanya bisa menangani *market order* karena fitur penutupan dan pembukaan kontrak diperuntukkan bagi pelaku pasar dengan nilai kontrak yang kecil.

1.5 Metodologi

Berikut adalah metodologi yang digunakan untuk pengembangan perangkat lunak ini, antara lain:

1. Melakukan studi literatur
 - Melakukan studi literatur terkait *runtime* Node.js.
 - Melakukan studi literatur terkait *front-end framework* VueJs, Tailwind.
 - Melakukan studi literatur terkait *back-end framework* Express.js.
 - Melakukan studi literatur terkait standar JSON Web Token.
 - Melakukan studi literatur terkait mengenai *web service*.
2. Melakukan analisis kebutuhan perangkat lunak berbasis web
 - Mempelajari cara kerja API Bybit.
 - Melakukan analisis terhadap data yang didapat dari API Bybit.

3. Merancang perangkat lunak
 - Membuat rancangan basis data untuk keperluan perangkat lunak
 - Membuat rancangan antarmuka perangkat lunak
 - Membuat *data flow diagram level 2*
 - Membuat rancangan *endpoint API*
4. Membangun perangkat lunak
5. Menguji perangkat lunak
 - Menguji fungsionalitas seluruh tombol dan menu dari perangkat lunak
 - Menguji fitur-fitur dari perangkat lunak dengan bantuan *remote access*
6. Menulis dokumen skripsi

1.6 Sistematika Pembahasan

Sistematika pembahasan pada Pembangunan Perangkat Lunak Cryptocurrency Berbasis Web dengan Memanfaatkan API Bybit Menggunakan Tailwind dan Node.js adalah sebagai berikut.

Bab 1 memuat latar belakang, rumusan masalah, tujuan, dan batasan masalah untuk pembangunan perangkat lunak berbasis web memanfaatkan API Bybit.

Bab 2 memuat landasan teori yang berkaitan dengan perancangan aplikasi, yang digunakan untuk mendukung aplikasi yang dibangun. Bab ini berisi mengenai *cryptocurrency*, bursa perdagangan *cryptocurrency*, API bursa Bybit, pemrograman berbasis web, dan *tech stack* yang digunakan pada perancangan aplikasi yaitu Express.js, Vue.js, Node.js, JSON Web Token, Tailwind, dan Chart.js.

Bab 3 memuat analisis masalah, analisis API Bybit, analisis kebutuhan dan spesifikasi perangkat lunak, analisis arsitektur perangkat lunak, analisis perangkat lunak untuk pembangunan perangkat lunak berbasis web memanfaatkan API Bybit.

Bab 4 memuat perancangan perancangan basis data, perancangan DFD Level 2, perancangan antarmuka, dan perancangan *endpoint* untuk pembangunan perangkat lunak berbasis web memanfaatkan API Bybit.

Bab 5 memuat implementasi, lingkungan implementasi, implementasi basis data, implementasi basis data, dan pengujian fungsional beserta pengujian eksperimental untuk pembangunan perangkat lunak berbasis web memanfaatkan API Bybit.

Bab 6 memuat kesimpulan dan saran untuk pembangunan perangkat lunak berbasis web memanfaatkan API Bybit.