

# BAB 1

## PENDAHULUAN

---

### 1.1 Latar Belakang

Uang saat ini menjadi alat tukar yang lumrah digunakan oleh semua orang. Perkembangan teknologi kini berkontribusi terhadap bagaimana suatu mata uang dapat digunakan baik itu sebagai alat pembayaran maupun sebagai alat tukar dapat dilakukan tidak hanya dalam kehidupan sehari-hari, namun kini transaksi seperti mengirim maupun menerima uang dapat dilakukan secara digital. *Cryptocurrency* (mata uang digital) atau mata uang *crypto* merupakan bentuk lain dari uang elektronik yang digunakan sebagai alat tukar. Pencatatan transaksi *cryptocurrency* (mengirim/menerima) mengadaptasi buku besar terdistribusi (*distributed ledger*), serta mengadaptasi sistem kriptografi untuk memverifikasi dan mengamankan setiap transaksi dalam sistem yang terdesentralisasi [1]. *Bitcoin*, *Etherium*, dan *Dash* merupakan salah satu dari banyaknya representasi mata uang krypto yang didukung dengan inovasi teknologi *blockchain*. Nilai *cryptocurrency* cenderung lebih tinggi jika ditukar ke mata uang konvensional. Contohnya saja 1 *Dash Coin* setara dengan Rp.2.220.000 pada Juli tahun 2019, dan nilai ini bisa saja bertambah atau berkurang seiring waktu [2]. *Cryptocurrency* dan *blockchain* merupakan dua hal yang saling berkaitan, karena *cryptocurrency* merupakan salah satu contoh sukses dalam penerapan teknologi *blockchain*. Suatu sistem yang dapat menyimpan seluruh transaksi serta data/informasi transaksi yang telah tersimpan tersebut dapat diakses oleh semua orang (yang berada dalam jaringan), merupakan salah satu kemampuan dari teknologi *blockchain*. Tidak hanya itu, *blockchain* juga memiliki kelebihan lain yakni setiap transaksi yang telah tersimpan tidak bisa diubah sehingga menutup celah untuk upaya manipulasi [3].

Oleh karena itu, simulasi ini akan menggambarkan transaksi *cryptocurrency* dalam jaringan *masternode* yang memang masih berkaitan dengan konsep teknologi *blockchain*, mulai dari bagaimana alur setiap blok yang ada pada *masternode* tercipta serta bagaimana alur setiap transaksi yang terjadi pada jaringan *masternode*.

## 1.2 Rumusan Masalah

Adapun rumusan masalah pada simulasi ini adalah :

1. Bagaimana alur kerja *blockchain* pada *masternode* dalam transaksi *cryptocurrency*?
2. Bagaimana alur setiap transaksi yang terjadi pada jaringan *masternode*?

## 1.3 Tujuan

Berdasarkan permasalahan yang ada, maka tujuan dari pembuatan simulasi *masternode* pada *cryptocurrency* diantaranya :

1. Membuat simulasi yang dapat menampilkan bagaimana setiap transaksi terjadi.
2. Membuat simulasi yang dapat menjelaskan alur kerja *blockchain* pada jaringan *masternode* dalam kaitannya dengan penyimpanan riwayat transaksi ke dalam blok.

## 1.4 Batasan Masalah

Adapun batasan masalah yang terdapat pada simulasi *masternode* pada *cryptocurrency* adalah :

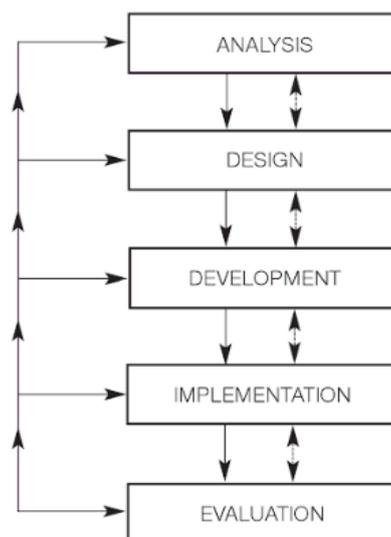
1. Simulasi ini hanya menampilkan transaksi *cryptocurrency* (mengirim uang/koin) dalam *masternode*, dan terdapat lima *node* lainnya yang berperan sebagai validator transaksi.
2. Simulasi ini hanya menampilkan satu *masternode* yang ada di dalam jaringan, dan besaran *share reward* nya senilai seratus koin.
3. Simulasi ini hanya dapat menampilkan satu proses transaksi mengirim uang/koin sampai bagaimana transaksi tersebut disimpan dan dilihat riwayatnya.

## 1.5 Definisi Operasional

Simulasi *masternode* pada *cryptocurrency* menggambarkan bagaimana transaksi *cryptocurrency* pada sistem *masternode* berjalan, serta seperti apa proses transaksi di dalamnya. *Cryptocurrency* erat kaitannya dengan blok-blok yang saling terhubung satu sama lain untuk menyimpan data transaksi keuangan yang terjadi. Sehingga dapat dikatakan bahwa *cryptocurrency* berjalan karena telah mengadaptasi teknologi *blockchain* [1]. Oleh karena itu, simulasi ini akan menampilkan bagaimana transaksi *cryptocurrency* berdasarkan sistem *masternode* yang telah mengadaptasi teknologi *blockchain*.

## 1.6 Metode Pengerjaan

Model pengembangan untuk simulasi ini adalah ADDIE model. ADDIE model merupakan akronim dari lima step/langkah pendekatan untuk mempelajari desain. Metode pengembangan ADDIE terdiri dari akronim *Analyze, Design, Development, Implementation, dan Evaluation*. Metode pengembangan ini menggunakan pendekatan yang akan memberikan arahan untuk mencapai hasil yang optimal karena dikerjakan secara sistematis dan bertahap [4].



**Gambar 1-1**  
**Metode ADDIE**

Berikut merupakan tahapan-tahapan dari model ADDIE [4]:

1. Analisis (*Analysis*)

Tahapan ini merupakan proses awal untuk mendefinisikan kebutuhan yang akan dipenuhi pada saat pengerjaan aplikasi. Proses yang dilakukan pada tahapan ini yakni pengumpulan data-data serta mendefinisikan data tersebut untuk digunakan pada tahap selanjutnya.

2. Desain (*Design*)

Tahapan ini berfokus pada desain pembuatan pemodelannya, seperti storyboard, skenario dan lain-lain.

3. Pengembangan (*Development*)

Tahapan ini akan mengimplementasikan desain yang telah dibuat, untuk kemudian dibuat ke dalam suatu simulasi dengan rincian kegiatan seperti pembuatan teks, grafik, animasi, audio, dan lain sebagainya.

4. Implementasi (*Implementation*)

Tahapan ini berfokus pada bagaimana aplikasi yang telah dikembangkan dapat berjalan serta memastikan semua komponen yang terdapat di dalamnya berjalan sesuai dengan kegunaannya.

5. Evaluasi (*Evaluation*)

Tahapan ini dilakukan dengan tujuan untuk mengetahui apakah aplikasi yang dibuat sudah berjalan dengan semestinya atau belum. Jika terdapat kekurangan, maka akan dilakukan pengembangan kembali.

## 1.7 Jadwal Pengerjaan

**Tabel 1-1**  
**Jadwal Pengerjaan**

Aktifitas	2018												2019																							
	Agustus			September			Oktober			November			Desember			Januari			Februari			Maret			April			Mei			Juni					
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Analisis	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■																				
Desain									■	■	■	■	■	■	■	■																				
Pengembangan																	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■				
Implementasi																									■	■	■	■	■	■	■	■				
Evaluasi																													■	■	■	■	■	■	■	■