

CUSTOM INDIKATOR PELACAK POLA REVERSAL CANDLESTICK PADA METATRADER 4 MENGGUNAKAN BAHASA PEMROGRAMAN METAQUOTE LANGUAGE 4

*(CUSTOM INDICATOR REVERSAL CANDLESTICK PATTERN TRACKER ON
METATRADER 4 USING THE METAQUOTE LANGUAGE 4 PROGRAMMING
LANGUAGE)*

¹ Whisnumurti Adhiwibowo, ² Ali As'ari

^{1,2} Teknik Informatika, Fakultas Teknologi Informasi dan Komunikasi

Universitas Semarang

¹ whisnu@usm.ac.id, ² asari.a.cp@gmail.com

ABSTRACT

Transaction activities in the financial markets there are various ways to analyze and predict trends or price movements of financial instruments. One way to analyze price movements is by studying the candlestick pattern reversing the direction of the trend. The obstacle faced when analyzing using a reversal candlestick pattern is the number of patterns and rules that are held to determine a trend reversal pattern on the candlestick. Therefore we need a tool or indicator that can analyze and provide information on the trend reversal pattern on the candlestick. In dealing with these problems, the authors use the waterfall model system development method with stages including analysis, design, coding, and testing. And the authors try to collect related data so that it can be used as a basic foundation in building indicator applications based on the programming language MetaQuote Language 4. The purpose of this study is to design, design and create a custom indicator on MetaTrader 4 that can analyze the candlestick pattern reversing the direction of the trend. The results of this study are an indicator that can track reversal candlestick patterns and display information on trend reversal candlestick patterns that are found to be used as a reference for transaction analysis on financial markets.

Keywords : *Candlestick, Reversal Pattern, Custom Indicator, MetaTrader 4, MetaQuote Language*

ABSTRAK

Kegiatan transaksi pada pasar keuangan terdapat berbagai cara untuk menganalisa dan memprediksi tren atau pergerakan harga dari instrumen keuangan. Salah satu cara menganalisa pergerakan harga dengan mempelajari pola *candlestick* pembalikan arah tren. Kendala yang dihadapi pada saat menganalisa menggunakan pola *candlestick* pembalikan arah adalah banyaknya pola dan aturan yang dimiliki untuk menentukan suatu pola pembalikan arah tren pada *candlestick*. Oleh sebab itu dibutuhkan sebuah alat atau indikator yang dapat menganalisa dan memberikan informasi pola pembalikan arah tren pada *candlestick*. Dalam menangani masalah tersebut, penulis menggunakan metode pengembangan sistem model *waterfall* dengan tahapan meliputi analisa, desain, pengodean dan pengujian. Serta penulis berusaha mengumpulkan data-data yang berkaitan agar dapat digunakan sebagai landasan dasar dalam membangun aplikasi indikator berbasis bahasa pemrograman *MetaQuote Language 4*. Tujuan dari penelitian ini adalah mendesain, merancang dan membuat sebuah custom indikator pada *MetaTrader 4* yang dapat menganalisa pola *candlestick* pembalikan arah trend. Hasil dari penelitian ini adalah sebuah indikator yang bisa melacak pola *candlestick* pembalikan dan menampilkan informasi pola *candlestick* pembalikan arah tren yang ditemukan untuk digunakan sebagai acuan analisa transaksi pada pasar keuangan.

Kata Kunci : *Candlestick, Pola pembalikan, Indikator, MetaTrader 4, MetaQuote Language*

1. PENDAHULUAN

Kemampuan membaca grafik menjadi modal awal bagi orang yang ingin belajar atau menekuni dunia pasar modal. *Candlestick* merupakan salah satu bentuk visualisasi pada grafik. Kelebihan *candlestick* adalah dapat menggambarkan kondisi pasar melalui harga pembukaan, penutupan, harga terendah, dan harga tertinggi dalam satu batang yang utuh. Dalam penerapannya pada analisa teknikal melalui studi *candlestick* terdapat aturan pola yang disebut pola *candlestick (candlestick pattern)*. Pola-pola ini dapat memberikan informasi pada *trader* apakah sedang tren naik, turun, atau akan terjadi pembalikan arah. Pada penelitian ini penulis akan fokus pada pola pembalikan arah. Karena, pola ini menunjukkan penurunan minat dari *trend* utama. Dengan mengetahui penurunan minat dari tren utama merupakan penanda berakhirnya tren yang bisa mengindikasikan akan terjadi koreksi pada pasar.

MetaTrader 4 adalah aplikasi perangkat lunak yang dikembangkan oleh Metatrader Corporation. *MetaTrader* menyediakan layanan penghubung antara *trader* dan server dari pialang / broker. Di dalam *MetaTrader* sendiri terdapat berbagai alat (indikator) untuk kebutuhan analisa teknikal. Selain menggunakan indikator standar pada *MetaTrader*, *trader* bisa mengembangkan indikator sesuai kebutuhan.

2. LANDASAN TEORI

Indikator

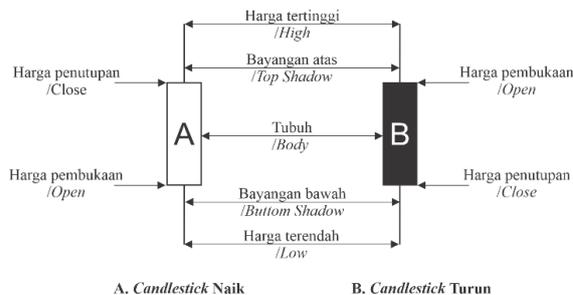
Indikator adalah alat yang dibuat berdasarkan perhitungan matematis atas harga, volume, atau minat pasar terhadap sebuah instrumen investasi atau kontrak keuangan. Secara umum, indikator merupakan suatu alat yang dapat memberikan informasi kepada *trader* dengan memetakan mengenai kondisi pasar yang terjadi saat ini serta melakukan prediksi mengenai kondisi pasar selanjutnya. Indikator memiliki fungsi untuk dapat memberikan pemetaan informasi kondisi pasar kepada *trader* sehingga dapat digunakan untuk menentukan strategi yang tepat untuk kondisi tersebut serta memberikan sinyal untuk masuk maupun keluar yang akan menuntun *trader* sehingga dapat masuk ataupun keluar dengan tepat [1].

Candlestick

Candlestick adalah sebuah teknik untuk memetakan pergerakan harga pasar baik itu saham, komoditi, maupun valuta asing. Teknik ini termasuk dalam kelompok analisa teknikal dan merupakan teknik

tertua yang diciptakan oleh Munehisa Homma pada tahun 1724 – 1803 di Jepang. Kemudian pada tahun 1991, Steve Nison memperkenalkan teknik ini ke dunia Barat dalam bukunya yang berjudul *Japanese Candlestick Charting Technique* [2].

Bagian-bagian dari *candlestick* dapat dilihat pada gambar 1.



Gambar 1. Bagian-bagian *Candlestick* [2]

Gambar 1 menunjukkan bagian-bagian yang membentuk *candlestick* terdiri dari harga pembukaan (*open*), harga tertinggi (*high*), harga terendah (*low*), harga penutupan (*close*), dan bayangan (*shadow*).

Pola Reversal Candlestick

Pola *reversal* adalah suatu pola yang mengisyaratkan akan adanya pembalikan arah tren. Jika pada saat tren naik atau tren turun kemudian pola ini muncul, maka akan diperkirakan harga akan bergerak berlawanan dengan arah tren sebelumnya [3].

Dikutip dari [3] macam-macam pola *reversal* yang terjadi pada *candlestick* adalah sebagai berikut :

a. Hammer

Pola ini muncul pada bagian tren menurun. Pola *Hammer* dapat dilihat pada gambar 2.



Gambar 2. Pola *Hammer*

Gambar 2 menunjukkan pola *hammer*. *Hammer* terbentuk dari *candlestick* tipe *long shadow*. Panjang *shadow* bawah minimal 2 kali panjang bodi. Panjang *shadow* atas tidak lebih dari setengah bodi *candlestick*. Warna bodi bisa hitam atau putih.

b. *Inverted hammer*

Inverted Hammer muncul pada tren menurun. Pola *Inverted Hammer* terlihat seperti pada gambar 3.

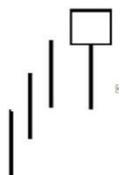


Gambar 3. Pola *Inverted Hammer*

Gambar 3 menunjukkan pola *inverted hammer*. *Inverted hammer* terbentuk dari *candlestick* tipe *long shadow*. Panjang *shadow* atas minimal 2 kali panjang bodi. Panjang *shadow* bawah hampir tidak terlihat atau tidak lebih dari setengah bodi *candlestick*. Warna bodi bisa hitam atau putih

c. *Hanging man*

Pola *hanging man* muncul pada tren naik. Pola *hanging man* dapat dilihat pada gambar 4.

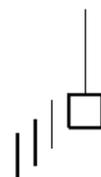


Gambar 4. Pola *Hanging Man*

Gambar 4 menunjukkan pola *hanging man*. *Hanging man* terbentuk dari *candlestick* tipe *long shadow*. Panjang *shadow* bawah minimal 2 kali panjang bodi. Panjang *shadow* atas hampir tidak terlihat atau tidak lebih dari setengah bodi *candlestick*. Warna bodi bisa hitam atau putih.

d. *Shooting star*

Pola *shooting star* muncul pada tren naik. Pola *shooting star* dapat dilihat pada gambar 5.

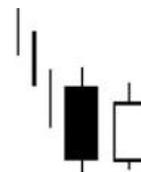


Gambar 5. Pola *Shooting Star*

Gambar 5 menunjukkan pola *shooting star*. *Shooting star* terbentuk dari *candlestick* tipe *long shadow*. Panjang *shadow* atas minimal 2 kali panjang bodi. Panjang *shadow* bawah hampir tidak terlihat atau tidak lebih dari setengah bodi *candlestick*. Warna bodi bisa hitam atau putih

e. *Piercing line*

Pola *piercing line* muncul pada tren menurun. Pola *shooting star* dapat dilihat pada gambar 6.



Gambar 6. Pola *Piercing Line*

Gambar 6 menunjukkan pola *piercing line*. Bodi *candlestick* putih dan *candlestick* sebelumnya hitam. Harga pembukaan *candlestick* putih lebih rendah dari harga penutupan *candlestick* hitam. Harga penutupan *candlestick* putih lebih dari setengah panjang bodi *candlestick* hitam dan tidak melewati pembukaan *candlestick* hitam.

f. *Dark cloud cover*

Pola *shooting star* muncul pada tren naik. Pola *shooting star* dapat dilihat pada gambar 7.



Gambar 7. Pola *Dark Cloud Cover*

Gambar 7 menunjukkan pola *dark cloud cover*. Pola ini muncul pada tren naik. Bodi *candlestick* putih dan *candlestick* sebelumnya hitam. Harga pembukaan *candlestick* hitam lebih tinggi dari harga penutupan *candlestick* putih. Harga penutupan *candlestick* hitam lebih dari setengah panjang bodi *candlestick* putih dan tidak melewati pembukaan *candlestick* putih.

3. METODOLOGI

Model pengembangan *waterfall* memiliki beberapa tahapan. Tahapan-tahapan pada model pengembangan *waterfall* dapat dilihat pada gambar 8.



Gambar 8. Model Pengembangan *Waterfall* [4]

Dari gambar 8 dapat dijelaskan tahapan-tahapan dari model *waterfall* adalah sebagai berikut:

a. Analisa kebutuhan

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan [4].

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean [4].

c. Pengodean

Desain ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program sesuai dengan desain yang telah dibuat pada tahap desain [4].

d. Pengujian

Pengujian fokus pada perangkat lunak dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan [4].

Dalam perancangan sistem ini menggunakan UML. Diagram yang digunakan penulis dalam pembangunan aplikasi berorientasi object, yaitu:

a. Use Case Diagram

Use case adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. Use case bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem dipakai [5].

b. Class Diagram

Class diagram mewakili pandangan statis dari suatu aplikasi. class diagram menggambarkan atribut, operasi, dan juga constraint yang terjadi pada sistem [5].

c. Activity Diagram

Activity diagram digunakan untuk menangkap tingkah laku dinamis dari sistem dengan cara menunjukkan aliran pesan dari satu aktifitas ke aktifitas lainnya [5].

d. Sequence Diagram

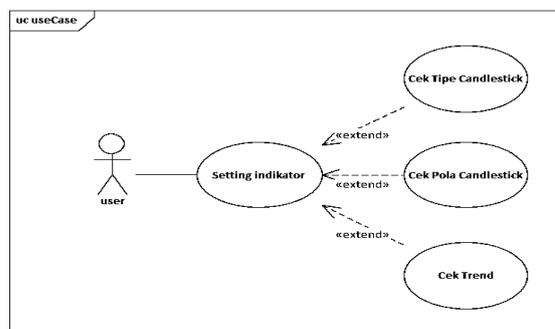
Diagram ini digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh objek dan pesan yang diletakkan diantara objek-objek di dalam use case [5].

4. HASIL DAN PEMBAHASAN

Perancangan

a. Use Case

Use case menggambarkan interaksi antara pengguna dan sistem. Diagram use case terlihat pada gambar 9.

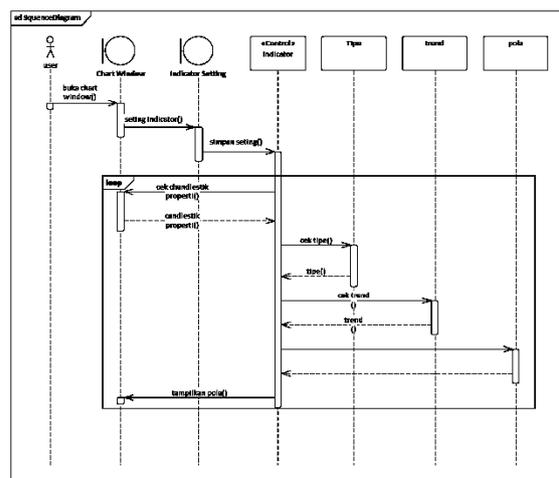


Gambar 9. Use Case sistem

Gambar 9 merupakan use case dari sistem. User melakukan pengaturan pada properti indikator. Setelah menyimpan pengaturan indikator akan dijalankan pada chart window. Indikator akan mengecek tipe menjalankan fungsi cek tipe candlestick, cek trend, cek pola.

b. Sequence Diagram

sequence diagram yang menggambarkan perilaku dan interaksi pada sebuah skenario. Diagram ini menunjukkan sejumlah objek dan pesan yang diletakkan diantara objek-objek di dalam use case. Sequence diagram dapat dilihat pada gambar 10.



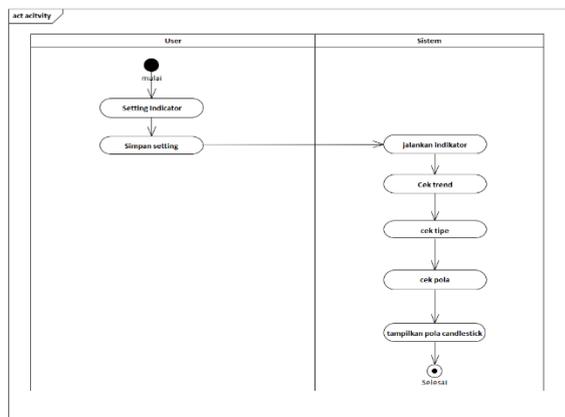
Gambar 10. Sequence Diagram

Gambar 10 menunjukkan user mengirim pesan ke sistem berupa pesan menjalankan indikator, sistem

akan menjalankan pesan yang dikirim dan menampilkan indikator.

c. Activity Diagram

Diagram ini menunjukkan aliran aktivitas dari. Diagram ini juga menggambarkan urutan aktivitas dari satu aktivitas ke aktivitas lainnya. *Activity diagram* dapat dilihat pada gambar 11.

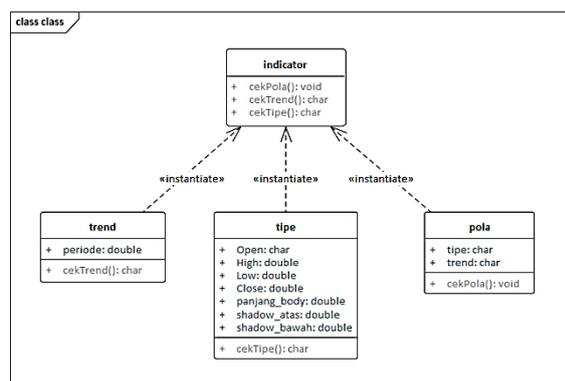


Gambar 11. Activity Diagram

Pada gambar 11 aktivitas yang digambarkan adalah *user* melakukan *setting* indikator dan menyimpannya. Kemudian, sistem akan melakukan aktivitas jalankan indikator, cek tipe, cek trend, cek pola dan menampilkan pola *candlestick*.

d. Class Diagram

Class diagram menggambarkan nama *class*, atribut, method, serta hubungan yang dimiliki. *Class diagram* dapat dilihat pada gambar 12.



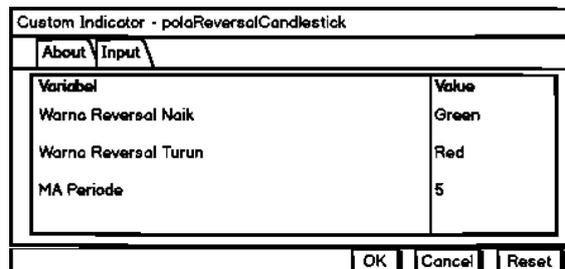
Gambar 12. Class Diagram

Pada gambar 12 digambarkan *class* tren dengan atribut periode dan *method* cek trend. *Class* tipe mempunyai atribut *open*, *high*, *low*, *close*, *panjang_body*, *shadow_atas* dan *shadow_bawah*. *Method* yang dimiliki *class* tipe adalah cek tipe. *Class* pola terdapat atribut tipe dan trend, serta memiliki *method* cek pola. *Class* indikator

merupakan *class instance* yang menjalankan *method* dari *class* tipe, tren, dan pola. *Method* yang terdapat pada *class* indikator adalah cek pola, cek trend, dan cek tipe.

Desain

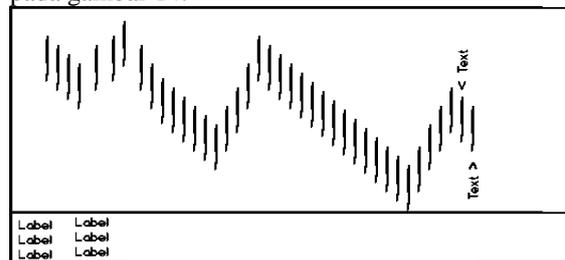
Desain tampilan setting indikator ditunjukkan pada gambar 13.



Gambar 13. Desain Tampilan *Setting* Indikator

Gambar 13 menunjukkan desain tampilan *setting* indikator. Pada desain *setting* indikator menampilkan pengaturan pada warna, dan periode *trend*.

Sedangkan desain tampilan indikator ditunjukkan pada gambar 14.



Gambar 14. Desain Tampilan Indikator

Gambar 14 merupakan desain tampilan indikator. Desain yang dibuat terdiri dari dua bagian. Bagian pertama merupakan jendela utama yang menunjukkan pola yang ditemukan pada grafik. Bagian kedua adalah jendela keterangan dari indikator.

Kode Program

a. Kode Program *Setting* Indikator

Kode program dari *setting* indikator ditampilkan pada gambar 15.

```

1. input color warnaIndikatorNaik =
   clrGreen; //Warna Reversal Naik
2. input color warnaIndikatorTurun =
   clrRed; //Warna Reversal Turun
3. input int periode = 5; //MA
   Periode
    
```

Gambar 15. Kode Program *Setting* Indikator

Pada gambar 15 nomor 1 dan 2 merupakan kode program untuk pengaturan warna pada tampilan indikator. Sedangkan nomor 3 merupakan kode program untuk periode trend.

b. Kode Program Indikator

Kode program dari indikator ditampilkan pada gambar 16.

```

1. CType candle;
2. CMATrend trendget;
3. CPola pola;
4. string trend =
   trendget.GetTrend(Symbol(), periode, 0,
   0, i, gClose1);
5. string tipe =
   candle.GetCandleType(gOpen1, gHigh1, gLow1, gClose1);
6. pola.pola_candle(trend, tipe, i,
   gOpen1, gClose1, gOpen1, gClose1,
   warnaIndikatorNaik,
   warnaIndikatorTurun);
7. ObjectCreate("Label1",
   OBJ_LABEL, 1, 0, 0);
   ObjectSetText("Label1", "HMN =
   Hanging Man :
   "+IntegerToString(hmn), 8, NULL, warnaIndikatorTurun);
   ObjectSet("Label1",
   OBJPROP_XDISTANCE, 10);
   ObjectSet("Label1",
   OBJPROP_YDISTANCE, 20);
    
```

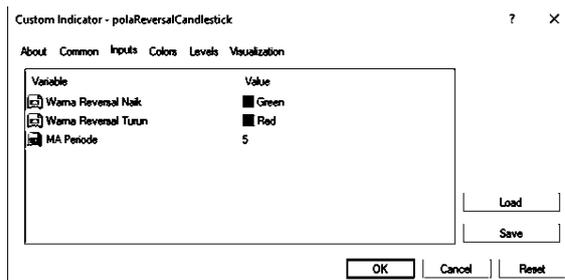
Gambar 16. Kode Program Indikator

Gambar 16 merupakan kode program dari indikator. Kode pada gambar 16 akan menjalankan fungsi cek tren, cek tipe, cek pola, dan menampilkan hasil pelacakan. Nomor 1 merupakan kode program untuk pemanggilan *class* CType. Nomor 2 menunjukkan kode program pemanggilan *class* CTrend. Nomor 3 menunjukkan program untuk memanggil *class* CPola. Nomor 4 merupakan pemanggilan fungsi method GetTrend yang berfungsi mengecek tren. Nomor 5 merupakan pemanggilan fungsi method GetTypeCandle yang berfungsi mengecek tipe *candlestick*. Nomor 6 merupakan pemanggilan method GetPola yang berfungsi menentukan pola pada *candlestick*. Kode program nomor tujuh berfungsi menampilkan informasi dari pola yang ditemukan.

Implementasi Program

a. Tampilan *Setting* Indikator

Implementasi dari tahapan desain dan pengodean *setting* indikator dapat dilihat pada gambar 17.

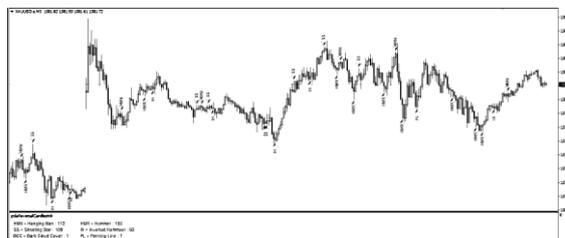


Gambar 17. Tampilan *Setting* Indikator

Pada pada gambar 17 tampilan *setting* indikator terdapat pengaturan pada warna indikator *reversal* naik, *reversal* turun dan *MA Periode* untuk pengaturan tren.

b. Tampilan Indikator

Implementasi dari tahapan desain dan pengodean indikator dapat dilihat pada gambar 18.



Gambar 18. Tampilan Indikator

Gambar 18 menunjukkan hasil dari desain dan pengodean untuk indikator. Pada tampilan indikator menampilkan pola yang ditemukan dan keterangan dari pola. Serta menampilkan pola jumlah pola yang ditemukan.

5. KESIMPULAN

Berdasarkan dari hasil implementasi dan pengujian aplikasi yang dibuat, maka dapat ditarik kesimpulan yaitu dengan adanya aplikasi indikator pelacak pola *candlestick* pembalikan arah tren mempermudah trader menganalisa pola *candlestick* pembalikan arah pada *chart window* *MetaTrader 4*. Jenis pola *candlestick* yang dapat dianalisa dengan aplikasi ini adalah pola *hammer*, *hanging man*, *inverter hammer*, *shooting star*, *dark cloud cover* dan *piercing line*.

Jenis pola *candlestick* pembalikan arah tren yang ditampilkan masih belum lengkap, untuk penelitian ke depan dapat dilengkapi dengan jenis-jenis pola *candlestick* pembalikan arah yang lain.

DAFTAR PUSTAKA

- [1] [Online]. Available:
www.analisaforex.com/indikator-forex-dan-apa-itu-fungsinya. [Diakses 12 Maret 2019].
- [2] F. T. Suharto, *Mengungkap Rahasia Forex*, Jakarta: Elex Media Komputindo, 2012.
- [3] [Online]. Available:
https://www.metaquotes.net/en/MetaTrader4/automated_trading/mql4. [Diakses 20 Desember 2019].
- [4] R. A. Sukamto dan M. Shalahuddin, *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*, Bandung: Informatika, 2016.
- [5] Munawar, *Analisis Perancangan Berorientasi Objek dengan UML (Unified Modeling Language)*, Bandung: Informatika, 2018.