

APLIKASI BANK TRANSAKSI DENGAN MENGAPLIKASIKAN ALGORITMA PENGESEHAN CAP JARI DAN PENYAHSULITAN KOD QR

Lee Kai Yi

Ravie Chandran A/L Muniyandi

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia

ABSTRAK

Perbankan atas talian, juga dikenali sebagai perbankan internet, adalah sistem pembayaran elektronik yang dibina bagi pelanggan bank melakukan transaksi kewangan seperti pemindahan wang, menyemak balance, melabur dan sebagainya melalui laman web institusi kewangan. Perkembangan perbankan dalam talian semakin pesat kerana ciptaan ini dapat menjimatkan masa dan tenaga dalam melakukan apa-apa transaksi. Pengguna boleh menyemak baki akaun dan memindahkan wang kepada akaun lain di mana-mana sahaja, tanpa pergi ke cawangan bank. Walau bagaimanapun, terdapat banyak penggodam yang sentiasa berusaha keras memecahkan sistem bank untuk mendapatkan maklumat pengguna sistem bank. Hal ini agak serius kerana ini boleh menyebabkan individu menghadapi kerugian yang sangat besar. Oleh itu, keselamatan sistem bank atas talian perlu ditingkatkan dari semasa ke semasa supaya pemilik akaun rasa selamat ketika berinteraksi dengan sistem bank atas talian. Dalam projek ini, algoritma pengesahan cap jari dan algoritma penyahsulitan kod QR akan diguna dan dianalisis bagi meningkatkan sistem keselamatan aplikasi bank. Penggunaan algoritma pengesahan cap jari dapat meningkatkan keselamatan sistem disebabkan keunikan cap jari bagi setiap individu. Dengan menggunakan alghoritma cap jari dalam sistem perbankan atas talian, keselamatan sistem boleh ditingkatkan dan masa untuk membuat transaksi dapat dijitamkan. Selain itu, penyahsulitan kod QR akan digunakan untuk membuat transaksi membeli-belah. Penggunaan kod QR dalam sistem pembayaran dapat mempercepatkan tansaksi dan amat selamat diguna.

1 PENGENALAN

Pengesahan cap jari adalah salah satu biometrik yang terkenal dan telah diperkenalkan di seluruh dunia ini. Setiap individu mempunyai cap jari yang unik. Cap jari boleh disalin tetapi memerlukan satu prosedur yang panjang dan penyalinan cap jari tanpa keizinan pemilik adalah diharam. Disebabkan keunikan dan konsistensinya dari masa ke semasa, cap jari telah digunakan untuk mengenalpastikan identiti seseorang pada era ini. Baru-baru ini, pengesahan cap jari telah dijalankan secara automatik, iaitu biometrik, akibat kemajuan dalam sektor pengkomputeran. Penggunaan cap jari sebagai kaedah mengenal pasti individu telah digunakan sejak abad ke-19 ketika Sir Francis Galton menerangkan beberapa perkara atau ciri-ciri yang

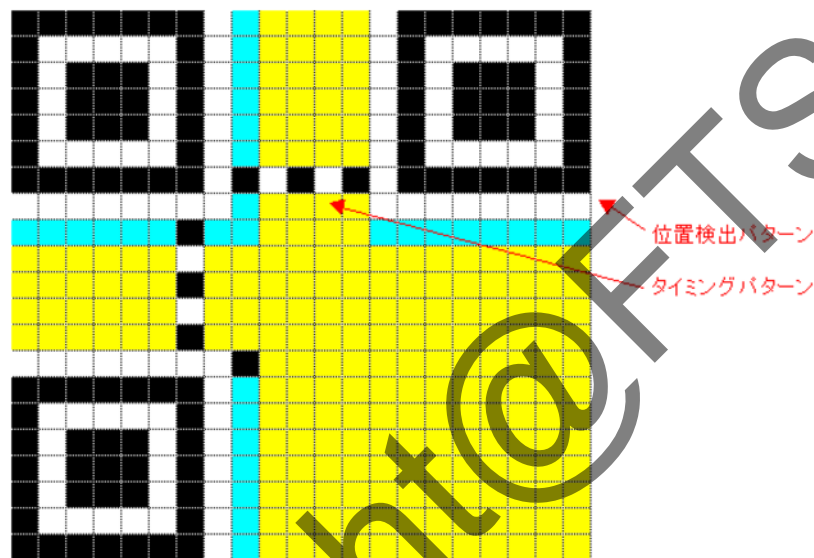
boleh diketahui oleh cap jari. "Galton Points" ini merupakan asas untuk sains pengenalan cap jari, yang telah berkembang dan beralih dari abad yang lalu. Pengesahan cap jari memulakan peralihannya ke automasi pada akhir tahun 1960-an bersama dengan kemunculan teknologi pengkomputeran. Dengan penciptaan komputer, salah satu subset daripada "Galton Points" telah digunakan untuk membangunkan teknologi cap jari automatik. (Mayhew, 2012)

Penggunaan pengesahan cap jari boleh meningkatkan keselamatan sistem komputer dan teknologi maklumat secara tidak langsung disebabkan keunikan. Misalnya aplikasi banyak menggunakan teknik pengesahan cap jari untuk mengesan penipuan dalam sesebuah transaksi kerana proses pengesahan ini adalah proses yang mengesahkan bahawa pengguna adalah siapa yang mereka nyatakan. Dalam algoritma pengesahan cap jari, identiti pengguna biasanya akan disimpan dalam beberapa jenis pengenalan, misalnya ID pengguna, nama pengguna atau kad. Algoritma pengesahan cap jari akan mengambil ID pengguna dan berbanding dengan cap jari baru dengan cap jari yang dikaitkan dengan ID pengguna sebelum ini. Jika kedua-dua cap jari dijumpai sama maka pengguna dibenarkan akses. Dalam pengesahan cap jari, perisian biometrik perlu mencari hanya rekod tunggal. (Thakkar, 2016) Hanya cap jari yang bersepadan dengan ID cap jari yang bersepadan dengan ID cap jari dalam rekod dapat mengakses data. Oleh itu, penggunaan algoritma pengesahan cap jari dalam sistem perbankan atas talian maya dapat meningkatkan keselamatan sistem tersebut.

Kod QR (*Quick Response*) merupakan nama tanda dagangan bagi sistem kod bar yang dicipta oleh Denso Wave, anak syarikat Toyota pada tahun 1994. Pada asalnya, kod QR digunakan untuk mengesan kenderaan dalam process pemasangan dengan pantas. Walaupun Denso Wave merupakan pencipta Kod QR ini, tetapi mereka telah memberikan lesen percuma kepada sesiapa sahaja menggunakannya dengan memaparkan teknologi baru ini ke atas internet pada ketika itu.

Kod QR bukan sahaja dapat memegang 100 kali lebih banyak data daripada kod bar 1 dimensi (1D), ia juga boleh diimbias oleh teknologi digital seperti kamera telefon bimbit. Blok hitam dan putih yang agak kecil dalam kod QR dibaca oleh sensor imej telefon bimbit, selapas itu ditafsirkan oleh pemproses sistem. Proses untuk membaca kod QR ini dikenali sebagai algoritma penyahsulitan kod QR. Tiga segi empat besar dalam kod QR bertindak sebagai sasaran penjajaran, sementara dataran yang lebih kecil di sudut yang lain bertindak untuk menormalkan saiz dan sudut pukulan. Seperti yang anda dapat lihat dari Rajah 1.1, jalur

biru berhampiran dataran penjajaran mengandungi maklumat pemformatan, dan kawasan kuning yang tinggal adalah data sebenar yang ditukar menjadi kod binari dan diperiksa untuk ralat sebelum dipaparkan. Data yang dikodkan boleh ditafsirkan sebagai salah satu daripada empat mod utama-angka, abjad angka, bait / binari, dan Kanji. Bentuk data lain juga boleh dipaparkan dengan sambungan yang sesuai.(Andrew Tarantola 2012) Dalam projek ini, algoritma penyahsulitan kod QR akan digunakan untuk melakukan transaksi membeli-belah. Proses transaksi dapat dimudahkan dengan menggunakan algoritma ini.



Rajah 1.1 Imej Penerangan Kod QR (Sumber diambil daripada (Andrew Tarantola 2012))

2 PENYATAAN MASALAH

Perkembangan dalam sector teknologi maklumat bukan sahaja memudahkan kehidupan masyarakat malahan telah meningkatkan ekonomi sesebuah negara secara tidak langsung. Pada masa ini, ekonomi digital menyumbang 17.8 peratus kepada ekonomi Malaysia dan disasarkan pada 20 peratus menjelang 2020. (RAHMAN, 2017)

Walaubagaimanapun, ada segelintir manusia telah menyalahgunakan perkembangan teknologi untuk mendapatkan “wang cepat” tanpa berusaha. Mereka menggunakan apa-apa cara sahaja untuk mendapatkan bank maklumat pemilik tanpa keizinan dan melakukan

transaksi atau mengeluarkan wang dari mesin atm. Hal ini menjadi lebih popular disebabkan perkembangan dalam pembinaan aplikasi bank mudah alih yang baru dalam seluruh dunia. “Perompak” sentiasa menumpukan perhatian terhadap masyarakat yang menggunakan aplikasi sistem perbankan atas talian untuk mendapatkan ID dan kata laluan mereka. Apabila “Perompak” tersebut berjaya mendapatkan ID dan kata laluan pemilik akaun, mereka akan menggunakan akaun mereka memindahkan wang kepada sesuatu akaun palsu dan mendapatkan wang daripada akaun palsu tersebut.

Pada masa sekarang, ramai masyarakat Malaysia masih menggunakan wang kertas untuk membeli-belah. Penggunaan wang kertas dalam pembeli-belahan boleh mendorong keselamatan seseorang individu terjejas kerana mereka akan menjadi mangsa salah seorang perompak yang telah mengawas pergerakan beberapa hari. Walaupun pengguna boleh menggunakan kad kredit atau kad debit mereka untuk membeli barang, tetapi cara ini tidak dapat menyelesaikan masalah tersebut dengan sepenuhnya. Akaun pemilik kad akan menjadi bahaya apabila pengguna tidak perasan kad mereka hilang.

3 OBJEKTIF KAJIAN

Objektif kajian ini adalah untuk:

- a) Meningkatkan keselamatan aplikasi mudah alih sistem perbankan yang sedia ada semasa log masuk dan membuat transaksi.
- b) Membina aplikasi perbankan atas talian maya yang mengaplikasikan algoritma pengesahan cap jari dan algoritma penyahsulitan kod QR.
- c) Mengeksploitasi fungsi “*Google Firebase*” seperti pendaftaran pengguna, penyimpanan dan pengambilan data.

4 METHOD KAJIAN

Projek ini akan menggunakan metodologi *spiral* yang menumpukan kepada skala yang kecil pada peringkat awal, kemudian mengenal pasti risiko, membuat plan untuk mengendalikan risiko tersebut dan menentukan sama ada ingin membuat langkah projek yang berikutnya ataupun iterasi yang sama sekali lagi. Kaedah *Spiral* ini mewarisi metodologi *waterfall* dengan

memperkenalkan prototaip. Pengurangan risikonya banyak digunakan dalam projek-projek yang besar. Melalui kaedah ini, risiko dalam sistem perbankan atas talian dapat dikenalpasti dan keselamatan sistem dapat ditingkat.

Terdapat beberapa fasa yang bakal dijalankan dalam projek ini. Antaranya adalah:

1. Fasa Konsep Keperluan (*Concept of Requirement*)

Keperluan fungsi baru, iaitu penggunaan cap jari pengesahan algoritma, akan ditakrifkan sebanyak mungkin dalam fasa ini. Peringkat ini akan melibatkan temuduga dengan beberapa pengguna sama ada pengguna luaran atau dalaman menggunakan aplikasi mudah alih sistem perbankan atas talian.

2. Fasa Konsep Operasi (*Concept of Operation*)

Reka bentuk awal dicipta untuk sistem perbankan atas talian dan cara perjalanan sistem di kalangan pengguna yang menggunakan aplikasi sistem perbankan atalian yang tertentu.

3. Fasa Pelan Keperluan (*Requirements Plan*)

Syarat-syarat yang digunakan supaya dapat membangunkan aplikasi sistem perbankan atas talian.

4. Fasa Prototaip 1 (*Prototype 1*)

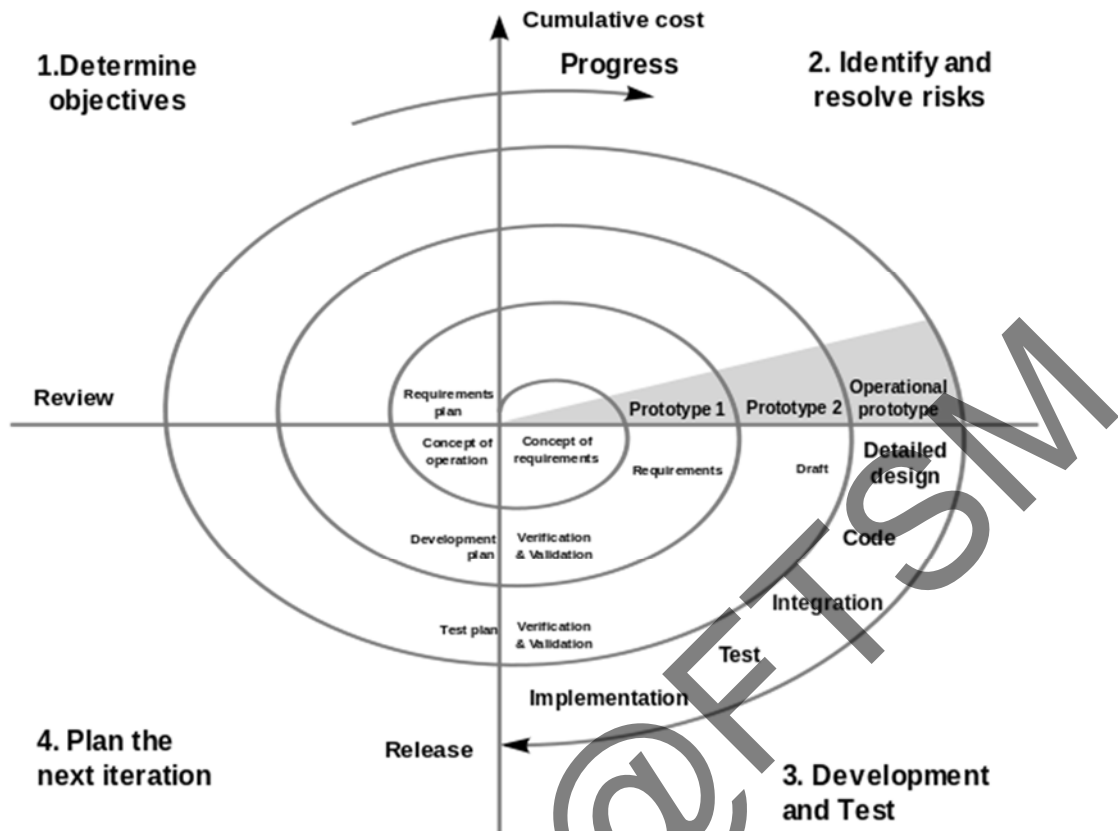
Satu prototaip aplikasi sistem perbankan atas talian yang mempunyai teknologi cap jari pengesahan akan dibina dan dicuba guna oleh beberapa pengguna aplikasi tersebut.

5. Fasa Reka Bentuk Terperinci

Fasa ini melibatkan perekaan sistem dan pangkalan data dengan merujuk kepada maklumat yang didapati berdasarkan aplikasi yang sedia ada. Antara muka aplikasi akan direka dalam fasa ini.

6. Fasa Ujikasi Pengkodan dan Pelaksanaan (*Code Test & Implementation*)

Kod dan algoritma aplikasi yang telah ditambahbaik akan cuba dijalankan sampai tidak ada kesilapan dan cuba digunakan oleh pengguna aplikasi sistem.



Rajah 1.2 Contoh Konsep Metodologi Spiral

5 HASIL KAJIAN

Bahagian ini membincangkan hasil daripada proses pembinaan aplikasi bank transaksi. Aktiviti pengkodan merupakan salah satu proses utama dalam membangunkan sistem. Pembangunan sistem perlu menepati keperluan sistem dan juga keperluan pengguna untuk membina satu sistem yang cekap. Untuk membangunkan sistem dengan lancar, beberapa perisian telah digunakan.

5.1 SISTEM PENGOPERASIAN *ANDROID*

Android adalah sistem operasi berasaskan Linux yang direka terutamanya untuk peranti mudah alih skrin sentuh seperti telefon pintar dan komputer tablet. Sistem pengendalian telah berkembang banyak dalam 15 tahun terakhir bermula dari telefon hitam dan putih ke telefon pintar atau komputer mini terkini. Salah satu OS mudah alih yang paling banyak digunakan pada hari ini adalah android. Android telah ditubuhkan di Palo Alto of California pada tahun 2003.(TARUN AGARWAL 2015)

Sesiapa sahaja boleh memuat turun SDK (*Software Development Kit*) dan menulis aplikasi untuk peranti mudah alih Android dan mula membangunkan untuk gedung Google Play. Pemaju yang menjual aplikasi di pasaran Google Play dikenakan caj kira-kira 30% daripada harga jualan mereka dalam yuran yang akan mengekalkan pasaran Google Play. Disebabkan *Google* telah membangunkan *Android*, banyak aplikasi yang berdasarkan *Google Service* seperti *Gmail*, *Google Calender*, *Google Maps* dan sebagainya dapat diguna dan dimuat turunkan oleh pengguna *Android*. Walaobagaimanapun, *Android* boleh diubahsuai oleh pembekal perantinya. Misalnya, ada segelintir peranti mudah alih *Android* menggunakan *Bing* sebagai enjin carian lalai dalam peranti mereka.(MARZIAH KARCH 2017)

API (*“Application Programming Interface”*) adalah satu pengantara yang membolehkan dua aplikasi berinteraksi antara satu sama lain. Aliran prosesnya boleh dijelaskan dengan menggunakan aplikasi *“Facebook”*. Apabila aplikasi *“Facebook”* dibuka atau digunakan dalam peranti mudah alih, ia akan menyambung ke internet dan menghantar data ke server *“Facebook”*. Selepas itu, server *“Facebook”* akan membaca dan menafsirkan data yang diterima dan melakukan reaksi yang diperlukan hantar balik kepada telefon pengguna. Aplikasi *“Facebook”* kemudian menerima dan menafsirkan data yang dihantar balik daripada server dan menunjukkan data tersebut kepada pengguna aplikasi. Inilah penggunaan *API*.(MuleSoft n.d.)

Dalam aplikasi ini, minima versi *API* peranti mudah alih pengguna yang diperlukan adalah 21, iaitu *“Lollipop”*. Peranti mudah alih yang mempunyai *API* yang kurang daripada *“Lollipop”* tidak boleh menggunakan aplikasi ini.

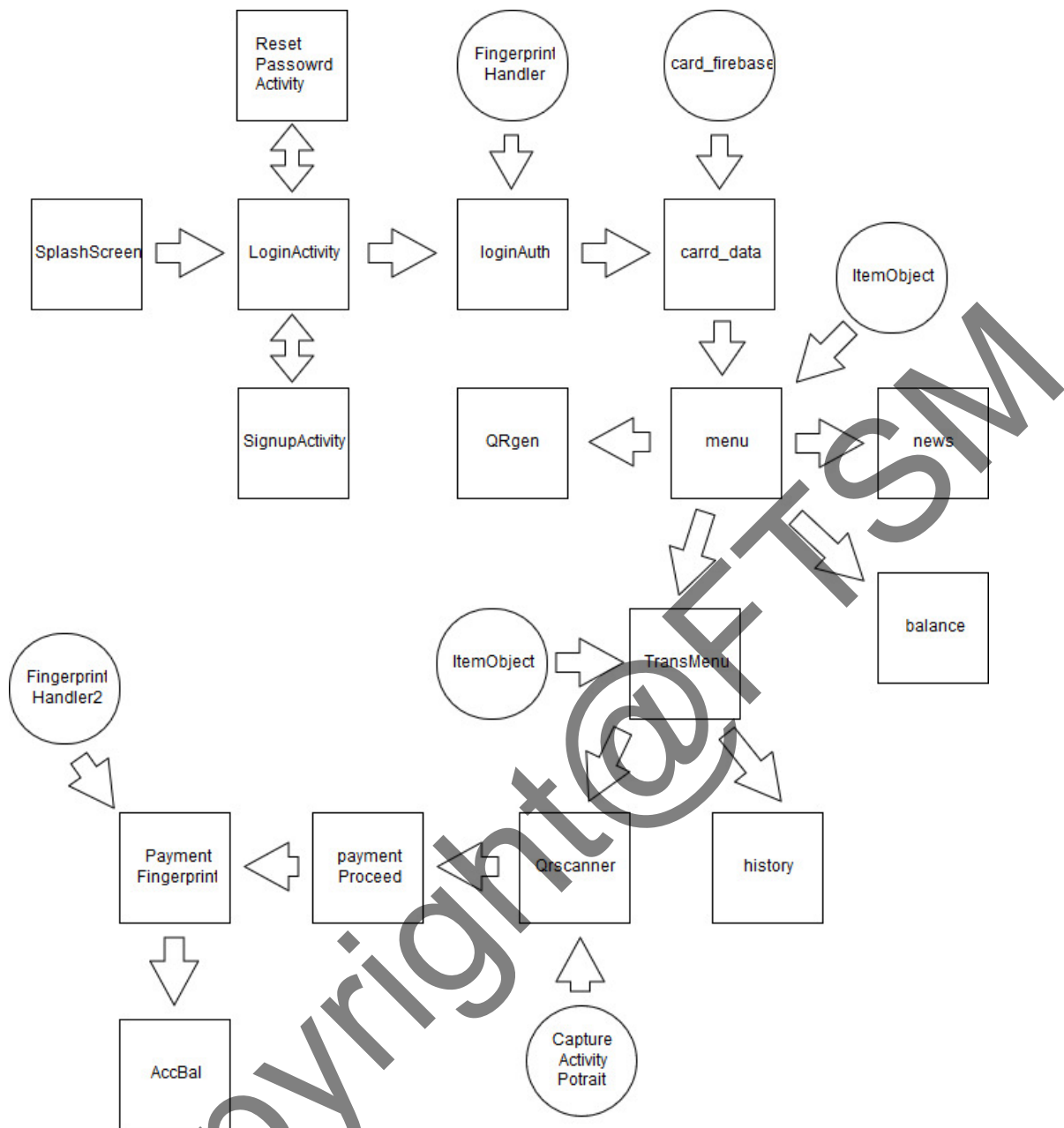
5.2 APLIKASI ANDROID STUDIO

Android Studio adalah satu aplikasi *integrated development environment(IDE)* rasmi bagi *Google's Android Operating System*. Aplikasi ini dibangunkan dengan kerjasama antara *Google* dengan *JetBrains*, pencipta salah satu *Java IDE* yang paling maju dan digunakan sampai masa sekarang. Aplikasi ini menggunakan satu sistem binaan projek yang baru, iaitu *Gradle*, yang menyediakan fleksibiliti, rasa bina yang sesuai dan sebagainya. Sistem binaan ini membolehkan penggunaannya membina projek dalam *IDE* serta server integrasi secara berterusan. (Xavier Ducrohet et al. 2013)

Dalam projek ini, aplikasi *Android Studio* digunakan untuk membina aplikasi ABT yang mempunyai fungsi pengesahan cap jari dan membuat pembayaran dengan penyahsulitan kod QR. Setiap projek *Android Studio* akan disimpan dalam computer pengguna format fail. Projek pembinaan aplikasi bank transaksi ini dinamakan sebagai "FYP_Demo". Setiap kelas yang dibina dalam projek ini akan di simpan dalam direktori fail seperti berikut :

```
"\FYP_Demo\app\src\main\java\com\example\leekaiyi\fyp_demo"
```

Dalam direktori tersebut, terdapat 21 java fail yang digunakan untuk membina fungsi aplikasi bank transaksi ini. Rajah 2.1 dibawah menunjukkan penggunaan kelas-kelas yang berada dalam projek "FYP_Demo" ini.



Rajah 5.1 Cara penggunaan 21 kelas dalam projek “FYP_Demo” ini

5.3 ALGORITMA PENYAHSULITAN KOD QR

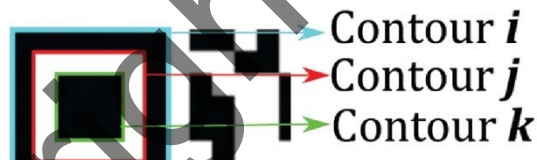
Kod QR (“*Quick Response*”) merupakan kod peningkatan taraf daripada barcode yang asalnya mempunyai satu dimensi kepada 2-dimensi. Fungsi utama kod QR asalnya adalah digunakan dalam industri Jepun untuk menyimpan data. Barcode hanya boleh menyimpan lebih kurang maximum 20 data string manakala kod QR dapat menyimpan beratus kali ganda berbanding dengan barcode. Selain itu, seperti nama yang dicipta pada kod 2-dimensi ini, kod

QR ini boleh dipenyahsulitkan dengan cepat dan tepat. Oleh itu, kod QR menjadi semakin terkenal pada masa kini.

Dalam projek ini, satu pembaca kod QR akan dibina untuk membaca dan mendapatkan kod QR yang dibinakan oleh penjual dan melakukan pembayaran melalui aplikasi ABT. Demi meningkatkan taraf keselamatan transaksi, algoritma pengesanan cap jari telah digunakan setelah aplikasi ini mendapatkan harga barang dengan menggunakan pembaca kod QR.



Rajah 5.2 Contoh kod QR. (Sumber diambil daripada laman web www.google.com)



Rajah 5.3 *Finder Pattern (FIP)* (Sumber daripada laman web www.google.com)

Setiap kod QR mempunyai tiga *FIPs* “*Finder Patterns*” (Rajah 2.2). Ketiga-tiga *FIPs* itu terletak di sudut kod QR. Setiap *FIP* mempunyai tiga “*Contours*” yang boleh dikesan, contohnya *i*, *j* dan *k* yang ditunjukkan dalam Rajah 2.2. Data yang berada dalam kod QR biasanya boleh dikesan dengan menggunakan kamera yang sedia ada dalam peranti mudah alih. Proses penyahsulitan kod QR adalah proses sebaliknya proses penyulitan kod QR. Rajah 2.3 menunjukkan contoh algoritma penyahsulitan kod QR.

Algorithm :

//String1 is the input and output.

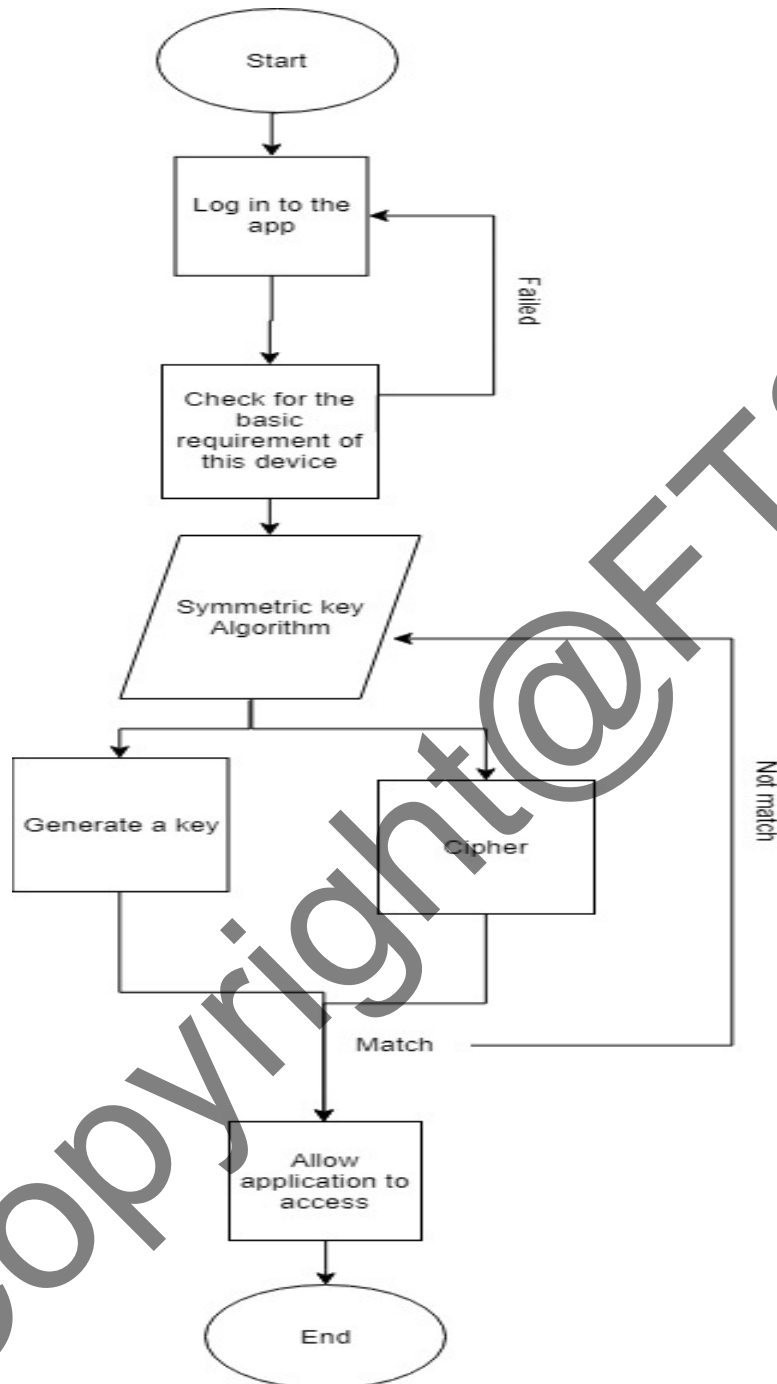
1. for each character in String1. do
 - Extract a character from the String1.
 - Convert this character into respective 8-bit length ASCII code.
 - Append this code to String2.//Initially String2 is empty.
- end for
2. Extract starting 8-bits from String2.
3. Convert this into number.
4. Remove _rst 8-bits from String2.
5. Remove 0's equal to (number-1) from the end of String2.
6. for each 6 characters in String2. do
 - Extract 6 character from the String2.
 - Convert it into equivalent code.
 - Store this character in String1.
- end for
7. Return String1.

Rajah 5.4 Contoh Algoritma penyahsulitan kod QR. (Sumber daripada laman web www.google.com)

5.4 ALGORITMA PENGESAHAN CAP JARI

Pengenalan teknologi pengesahan dan pengenalpastian cap jari telah meningkatkan keselamatan aplikasi dan sistem. Disebabkan keunikan cap jari bagi setiap individu, cap jari banyak digunakan dalam pengenalpastian seseorang idividu, kawalan akses dan sebagainya. Pengesahan cap jari digunakan dalam sistem yang berkaitan dengan kawalan akses untuk mengesahkan pengguna yang ingin mengakseskan aplikasi atau sistem adalah pemilik asal akaun tersebut. Dalam projek ini, pengesahan cap jari telah digunakan untuk mengenalpastikan pengguna yang log masuk adalah pemilik peranti mudah alih. Disebabkan setiap individu mempunyai cap jari yang berbeza dan unik, penggunaan cap jari ini adalah amat selamat digunakan untuk mengesahkan pengguna yang ingin mengakses aplikasi atau sistem. Rajah 2.3

menunjukkan algoritma pengesahan cap jari yang digunakan untuk mengesahkan akses pengguna.



Rajah 5.5 Algoritma Pengesahan Cap Jari (Sumber daripada www.google.com)

5.5 SEBAB PILIH CARA PENGEKODAN

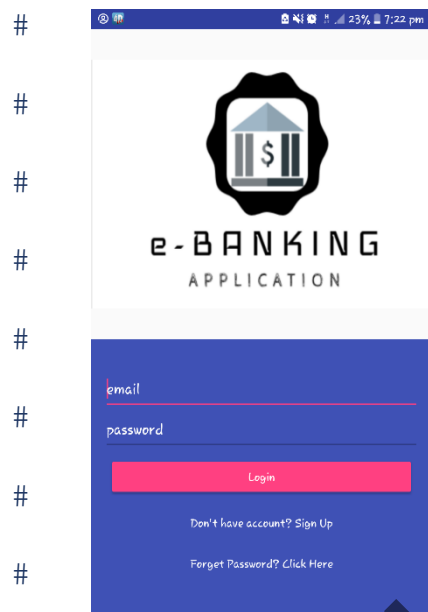
Dalam pembinaan aplikasi bank transaksi ini, terdapat beberapa fungsi kod dihasilkan dengan rujukan website www.github.com seperti kod dalam fail “FingerprintHanlder.java” , “FingerprintHandler2.java”, “Qrscanner.java” dan sebagainya. Sebab utama menggunakan kod yang dipaparkan dalam website www.github.com adalah kerana kebanyakan kod dalam website ini boleh dijalankan tanpa bermasalah. Selain itu, jenis kod bagi fungsi-fungsi yang ingin dicari senang untuk dicapai dalam website ini. Hampir semua pengeluar video tunjuk ajar pengekodan fungsi tertentu telah memuatnaikkan fail pengekodan mereka ke dalam website ini. Oleh itu, kod yang digunakan dalam pembinaan apikasi ini kebanyakan menggunakan sumber kod daripada www.github.com.

Dalam antara muka pendaftaran dan log masuk sistem, saya telah menukarkan bentuk butang dan fungsi butang yang sedia ada dalam “github”. Selepas itu, dalam algoritma pengesahan cap jari, saya telah mengubah suai kodnya dalam kod setelah pengguna berjaya mengesahkan cap jari mereka. Perkataan yang akan ditunjukkan, imej “tick” dan mengaplikasikan “Google Firebase” dalam kelas “FingerprintHandler.java” adalah barang yang saya tambahkan daripada sumber kod sedia ada. Dalam kelas “Qrscanner.java”, saya telah mengubah suaikan cara orientasi kamera yang didapati daripada laman web www.github.com. Selain itu, saya telah menyimpan hasil yang diperolehi selepas imbasan kod QR dan gunakan hasil tersebut dalam aktiviti lain seperti dalam kelas “paymentProceed.java”.

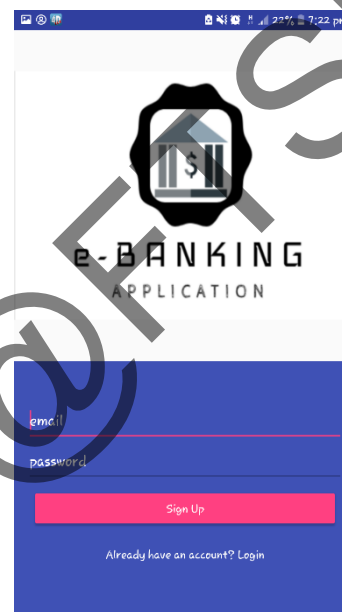
5.6 REKA BENTUK ANTARA MUKA

Antara muka adalah sangat penting dalam sebuah sistem kerana ia memberikan imperasi pertama semasa menggunakan aplikasi tersebut. Bukan itu sahaja, antara muka juga memainkan peranan sebagai medium interaksi sebuah sistem dengan penggunanya. Antara muka yang mesra dan senang difahami dapat menyebabkan pengguna memahami penggunaan sistem tersebut serta menggunakannya dengan mudah. Antara muka yang mesra penggunaan juga dapat memastikan tugas yang dikehendaki oleh pengguna dapat diselesaikan oleh sistem.

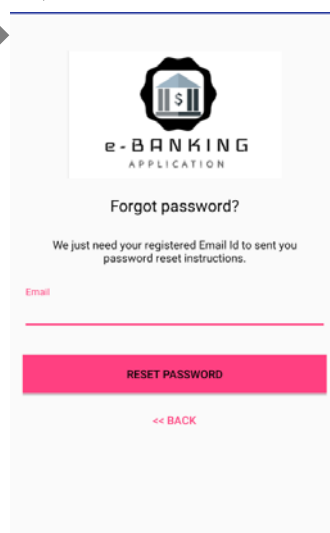
5.2.1 ANTARA MUKA LOG MASUK, DAFTAR DAN LUPA KATA LALUAN



Rajah 5.6 Antara muka log masuk

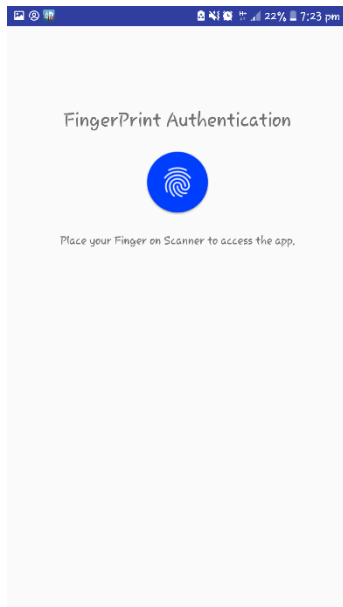


Rajah 5.7 Antara muka daftar

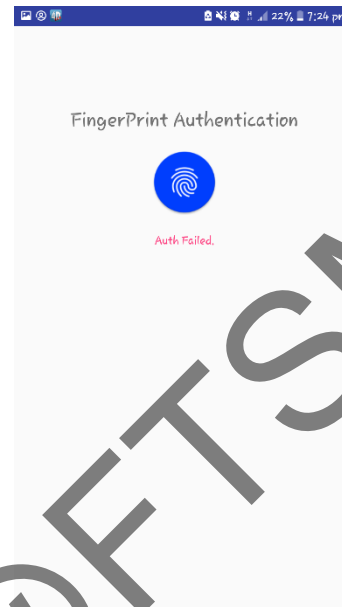


Rajah 5.8 Antara Muka lupa kata laluan

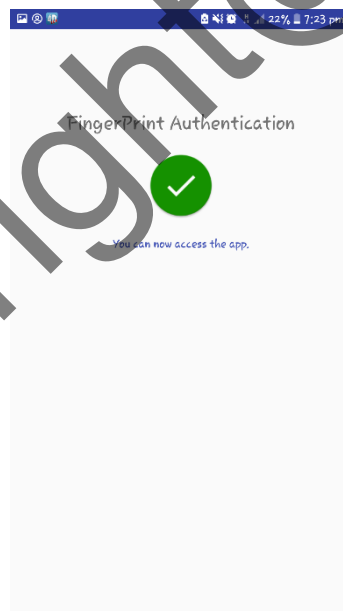
5.2.2 ANTARA MUKA MENGGUNAKAN PENGESAHAN CAP JARI UNTUK LOG MASUK KE APLIKASI



5.9 Antara muka pengesahan cap jari

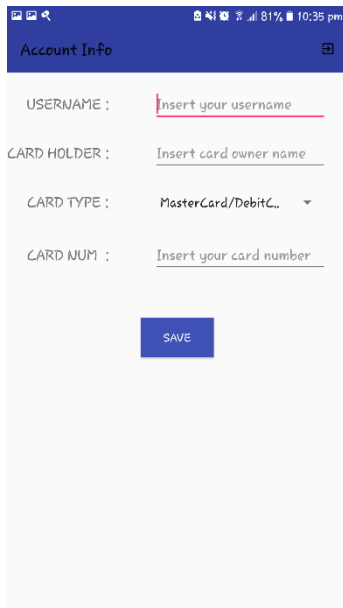


5.10 Antara muka gagal pengesahan

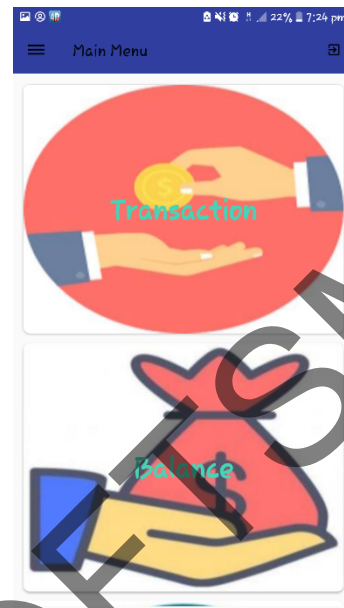


5.11 Antara muka berjaya membuat pengesahan

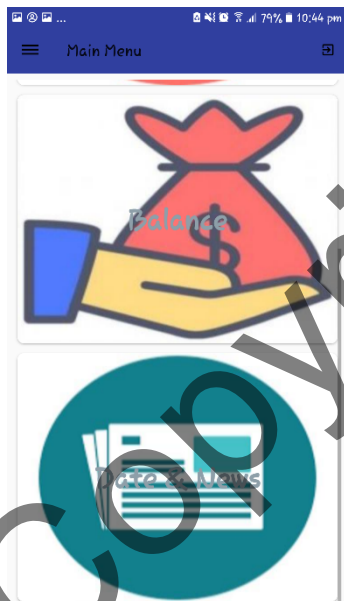
5.2.3 ANTARA MUKA APLIKASI SELEPAS LOG MASUK KE DALAM APLIKASI



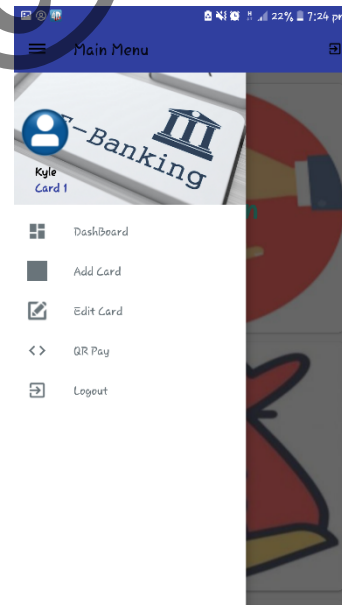
Rajah 5.12 Antara muka info akaun



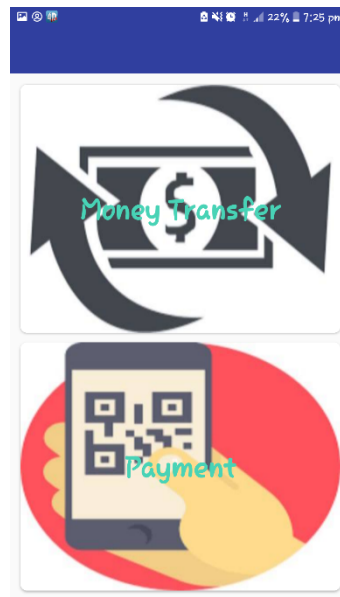
Rajah 5.13 Antara muka fungsi bank (1)



Rajah 5.14 Antara muka fungsi bank (2)



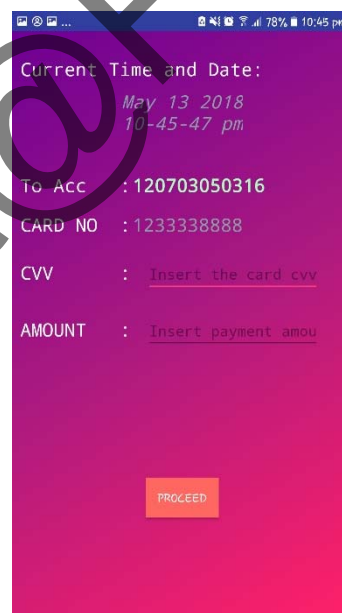
Rajah 5.15 Antara muka *side bar*



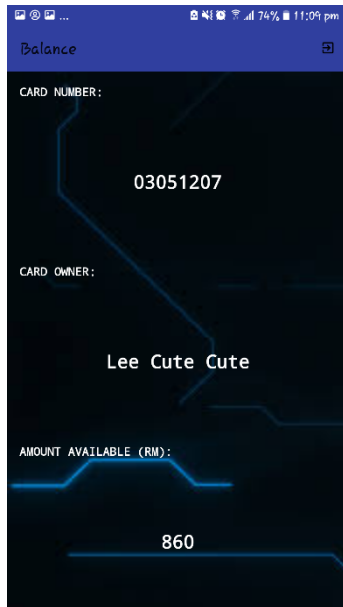
Rajah 5.16 Antara muka fungsi transaksi



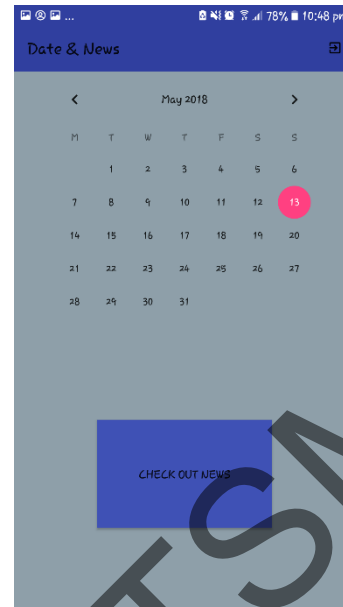
Rajah 5.17 Antara muka bayar dengan kod QR



Rajah 5.18 Antara muka pembayaran



Rajah 5.19 Antara muka baki akaun



Rajah 5.20 Antara muka berita baru

6 KESIMPULAN

Aplikasi Bank Transaksi yang telah dibina boleh meningkatkan keyakinan pengguna semasa melakukan sesuatu transaksi melalui aplikasi ini. Penggunaan algoritma cap jari dapat meningkatkan keselamatan sistem disebabkan keunikan cap jari yang ada pada setiap individu. Melalui aplikasi ini, pengguna boleh membuat bayaran dengan menggunakan kod QR. Penggunaan algoritma penyahsulitan kod QR memudahkan proses transaksi dan amat selamat digunakan. Selain itu, pengguna juga boleh menggunakan aplikasi ini untuk menyemak baki yang terdapat dalam bank akaun serta sentiasa memperoleh cerita baharu yang terbaru daripada pihak rasmi bank dengan menggunakan aplikasi ini.

Kesimpulannya, pembangunan sistem Aplikasi Bank Transaksi ini akan memenuhi objektif. Pengguna boleh menggunakan aplikasi ini untuk melakukan transaksi tanpa kerisauan. Tambahan pula, sistem ini berpotensi untuk diperkembangkan supaya menjadi aplikasi perbankan atas talian yang lebih sistematik dan selamat.

7 RUJUKAN

Andrew Tarantola. 2012. How QR Codes Work and Why They Suck So Hard. *18/12*.
<https://gizmodo.com/5969312/how-qr-codes-work-and-why-they-suck-so-hard> [30
March 2018].

MARZIAH KARCH. 2017. What Is the Android Operating System? *2 OCTOBER*.
<https://www.lifewire.com/what-is-google-android-1616887> [11 December 2017].

MuleSoft. (n.d.). What is an API? (Application Programming Interface) | MuleSoft.
<https://www.mulesoft.com/resources/api/what-is-an-api> [13 May 2018].

TARUN AGARWAL. 2015. What is Android? Introduction of Android OS & its
Applications. [https://www.elprocus.com/what-is-android-introduction-features-
applications/](https://www.elprocus.com/what-is-android-introduction-features-applications/) [11 December 2017].

WILL SIMONDS. 2014. Do mobile apps raise the risk of online bank fraud? *6TH OCTOBER*.
<https://www.abine.com/blog/2014/mobile-apps-and-online-bank-fraud/> [11 December
2017].

Xavier Ducrohet, Tor Norbye & Katherine Chou. 2013. Android Developers Blog: Android
Studio: An IDE built for Android. *15 May*. [https://android-
developers.googleblog.com/2013/05/android-studio-ide-built-for-android.html](https://android-developers.googleblog.com/2013/05/android-studio-ide-built-for-android.html) [11
December 2017].