

APLIKASI MUDAH ALIH CARIAN PENGASUH ‘MYKIDCARE’

Mohd Fazni Bin Mohd Salleh
Ts. Dr. Hasimi Sallehuddin

Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia

ABSTRAK

Aplikasi mudah alih MyKidCare merupakan sebuah platform yang memudahkan ibubapa terutamanya ibubapa yang memerlukan khidmat asuhan pada waktu yang fleksibel berdasarkan keperluan mereka. Aplikasi ini membolehkan para ibu bapa untuk membuat carian kepada individu yang menawarkan perkhidmatan asuhan berdasarkan lokasi pengasuh. Aplikasi ini menyediakan beberapa fungsi seperti membuat carian pengasuh, melihat profile pengasuh, membuat tempahan dan juga membuat penilaian kepada tahap kepuasan perkhidmatan pengasuh selepas menggunakan perkhidmatan mereka sebagai penanda aras bagi pengguna lain yang ingin mengambil khidmat pengasuh tersebut. Selain itu, aplikasi ini juga dapat menjana pendapatan kepada pengasuh yang menggunakan aplikasi ini. Admin perlu mengesahkan permohonan setiap permohonan untuk mendaftar menjadi pengasuh sebagai tapisan keselamatan. Aplikasi ini akan dibangunkan dengan menggunakan Android Studio dan MySQL untuk menyimpan data pengguna. Kesimpulannya, aplikasi mudah alih ini bertujuan untuk memudahkan ibubapa mencari khidmat asuhan dalam masa yang singkat dan pada masa yang sama aplikasi ini juga dapat menghasilkan pendapatan tambahan kepada individu sebagai pengasuh.

1. PENGENALAN

Aplikasi mudah alih adalah merupakan salah satu kemudahan yang disediakan oleh telefon pintar untuk membolehkan pengguna melakukan tugas harian. Boleh dikatakan hampir setiap generasi sekarang mempunyai terutamanya golongan yang berkerja menggunakan telefon pintar untuk berkomunikasi, melayari media sosial, menyimpan nota –nota penting, membuka dan membaca email dan sebagainya.

Perkembangan teknologi terkini yang pesat telah menyumbang kepada banyaknya perisian mudah alih dibangunkan bagi menyelesaikan masalah harian manusia. Antara sistem pengoperasian bagi telefon pintar yang terkenal adalah sistem pengoperasian Android yang dibangunkan oleh google dan juga IOS yang dibangunkan oleh Apple inc. Bagi pengguna

Android mereka boleh mendapatkan aplikasi mudah alih melalui aplikasi google play, manakala bagi pengguna produk Apple mereka boleh mendapatkan aplikasi mudah alih melalui aplikasi app store. Aplikasi mudah alih ini ada yang percuma dan ada yang berbayar. Ada juga aplikasi yang diberikan tempoh percubaan ataupun 'free trial' dan ada yang memberikan fungsi yang asas secara percuma tetapi jika ingin menggunakan fungsi-fungsi yang premium pengguna harus membeli aplikasi tersebut.

Aplikasi mudah alih menawarkan pelbagai kemudahan seperti perkhidmatan penghantaran makanan, e-hailing seperti grab car dan pelbagai jenis lain-lain perkhidmatan yang boleh memberikan kemudahan kepada pengguna serta dapat menjana sumber pendapatan yang agak lumayan. Antara perkhidmatan yang mendapat permintaan tinggi terutamanya di kawasan bandar ialah perkhidmatan pengasuh kanak-kanak. Kebanyakan pasangan suami isteri yang mempunyai kerjaya masing-masing tentulah memerlukan khidmat pengasuh untuk menjaga anak mereka ketika mereka keluar bekerja atau mempunyai hal-hal penting yang perlu diselesaikan yang menyebabkan mereka terpaksa mencari seseorang untuk menjaga dan menguruskan anak mereka.

2. PENYATAAN MASALAH

Ramai ibu bapa yang bekerja mengalami masalah untuk mendapatkan khidmat asuhan bagi anak mereka terutamanya mereka yang hanya ingin mengambil perkhidmatan asuhan mengikut waktu yang fleksibel berdasarkan keperluan mereka. Akan tetapi agak sukar untuk mendapatkan khidmat pengasuh dengan serta merta. Ada antara ibu bapa yang mengambil no telefon pengasuh yang diiklankan di mana-mana papan notis akan tetapi segala maklumat berkaitan pengasuh sukar untuk dikenalpasti kesahihannya, selain itu juga agak sukar untuk mengukur tahap kebolehan pengasuh dalam mengendalikan kanak-kanak dan juga agak berisiko jika mereka meninggalkan anak-anak mereka kepada individu yang kurang dikenali. Setiap ibu bapa pasti menitik beratkan isu keselamatan anak-anak dan mereka perlu berhati-hati di dalam memilih pengasuh anak-anak mereka.

3. OBJEKTIF

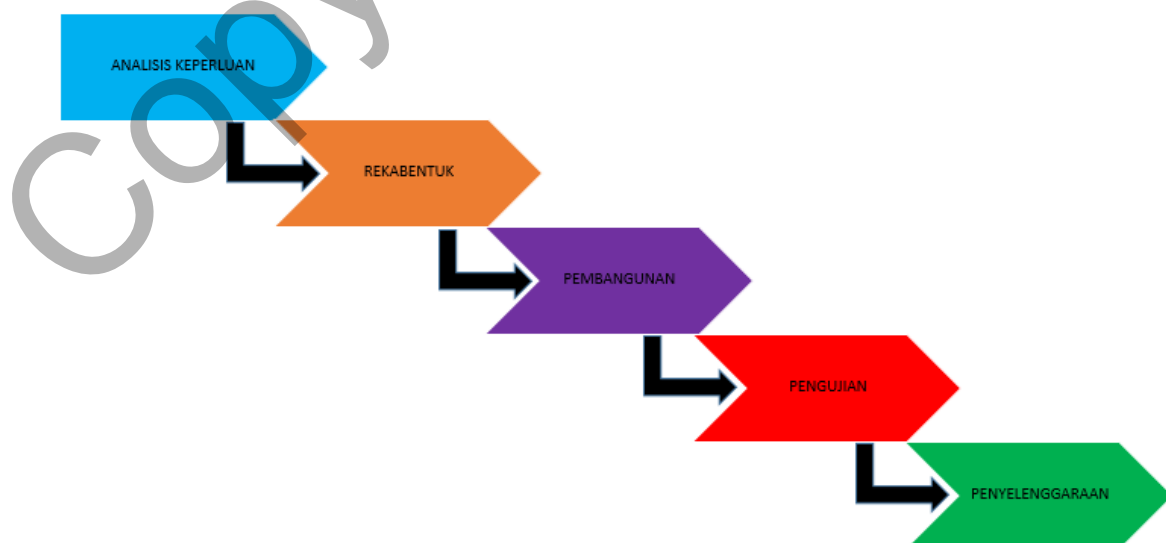
Objektif projek ini adalah seperti berikut :

- i. Menyediakan satu platform aplikasi mudah alih kepada ibu bapa untuk mencari perkhidmatan asuhan kanak-kanak secara atas talian.
- ii. Menyediakan satu platform aplikasi mudah alih kepada individu untuk menawarkan perkhidmatan asuhan kanak-kanak secara atas talian.
- iii. Memberikan lebih banyak pilihan kepada ibu bapa jika terdapat beberapa pengasuh yang berada dalam jarak kawasan liputan mereka.
- iv. Mampu menjana pendapatan terutamanya kepada golongan suri rumah dan sebagainya.

4. METODOLOGI

Metodologi yang digunakan dalam proses membangunkan aplikasi mudah alih MYKIDCARE ini adalah metodologi “Waterfall”. Dalam proses pembangunan yang menggunakan metodologi “Waterfall”, setiap fasa perlu diselesaikan terlebih dahulu sebelum fasa lain boleh bermula dan tiada pertindihan dalam fasa-fasa yang dilaksanakan.

Pendekatan metodologi ini dilaksanakan secara berturut-turut mengikut urutan fasa yang telah ditetapkan iaitu fasa analisis keperluan, fasa reka bentuk, fasa pembangunan, fasa pengujian dan fasa penyelenggaraan.



Rajah 1 Model Waterfall

i. Fasa Analisis Keperluan

Fasa ini merupakan fasa pertama yang akan memulakan pembangunan aplikasi menggunakan metodologi “Waterfall”. Fasa ini mengumpul semua idea dan aspek-aspek penting yang perlu ada dalam membina aplikasi ini. Fasa ini juga merupakan fasa yang akan mengumpulkan idea idea baru dan penambahbaikan terhadap aplikasi.

ii. Fasa Reka Bentuk

Fasa reka bentuk merupakan fasa di mana menentukan semua keperluan platform termasuk perisian yang akan digunakan dalam membangunkan aplikasi ini. Proses reka bentuk aplikasi daripada segi penampilan turut berjalan dalam fasa ini.

iii. Fasa Pembangunan

Fasa ini merupakan fasa membangunkan aplikasi berdasarkan segala idea yang telah diperoleh daripada fasa-fasa sebelum ini. Fasa ini melibatkan proses pengaturcaraan yang bakal dijalankan menggunakan perisian yang telah ditentukan. Dalam fasa ini, sebuah aplikasi akan dibangunkan sepenuhnya dan akan diuji di dalam fasa pengujian.

iv. Fasa Pengujian

Fasa pengujian ini merupakan satu fasa yang akan menguji aplikasi mudah alih yang telah dibangunkan. Pengujian akan dijalankan terhadap semua aspek yang ada pada aplikasi seperti fungsi yang terdapat dalam aplikasi dan sebagainya. Semua data dan keputusan yang diperoleh akan didokumentasikan dan di hantar ke fasa penyelenggaraan.

v. Fasa Penyelenggaraan

Melalui fasa penyelenggaraan, segala dokumentasi data, keputusan dan maklum balas yang diperoleh daripada fasa pengujian akan digunapakai dalam proses untuk melakukan penyelenggaraan dan penambahbaikan terhadap aplikasi mudah alih yang dibangunkan.

Aplikasi ini akan dibangunkan menggunakan aplikasi *Android Studio*. *Android Studio* merupakan perisian yang popular dikalangan pembangun aplikasi untuk sistem pengoperasian *Android*. *Android Studio* adalah persekitaran pembangunan terintegrasi

rasmi (IDE) untuk sistem operasi Android *Google*, dibina di atas perisian JetBrains 'IntelliJ IDEA dan direka khusus untuk pembangunan *Android*. Ianya boleh didapati untuk dimuat turun dan digunakan di sistem pengendalian berasaskan Windows, macOS dan Linux Sejak 7 Mei 2019, Kotlin adalah bahasa pengaturcaraan pilihan *Google* untuk pembangunan aplikasi *Android*. Namun, bahasa pengaturcaraan lain disokong oleh Android Studio, seperti Java dan C ++.

Berikut adalah spesifikasi minimum bagi perkakasan untuk android studio.

- Microsoft® Windows® 7/8/10 (64-bit)
The Android Emulator only supports 64-bit Windows.
- 4 GB RAM minimum, 8 GB RAM recommended
- 2 GB of available disk space minimum,
4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

Berikut adalah spesifikasi minimum bagi perkakasan pengguna (Telefon pintar)

- CPU: Quad Core CPU 1.0 GHz minimum
- RAM: 2 GB minimum
- OS: Android 4.0 or higher
- Storage space: 3 GB minimum

Node JS adalah merupakan sebuah perpustakaan dan persekitaran runtime merentas platform untuk membolehkan aplikasi JavaScript dijalankan di luar pelayar web. Ianya berasaskan sumber terbuka yang digunakan untuk membuat aplikasi JavaScript pada sisi pelayan. Rangka kerja(Framework) ini menawarkan perpustakaan yang kaya dengan pelbagai modul JavaScript untuk memudahkan proses pembangunan web.

Ionic adalah merupakan toolkit UI(Paparan Pengguna) sumber terbuka untuk membina aplikasi mudah alih dan desktop berkualiti tinggi menggunakan teknologi web (HTML, CSS, dan JavaScript/TypeScript). Rangka Kerja Ionik difokuskan pada pengalaman pengguna(UX) pada bahagian hadapan (frontend), atau interaksi UI aplikasi (kawalan, interaksi, gerak isyarat, animasi). Ianya mudah untuk dipelajari, dan

diintegrasikan dengan baik dengan perpustakaan atau rangka kerja lain seperti Angular, ataupun secara sendiri tanpa rangka kerja frontend menggunakan skrip mudah yang tersedia.

VSCode atau Visual Studio Code adalah kod editor yang boleh digunakan dengan pelbagai bahasa pengaturcaraan, termasuk Java, JavaScript, Go, Node.js dan C++.

MySQL adalah merupakan sistem pengurusan pangkalan data yang akan digunakan untuk menguruskan pangkalan data bagi aplikasi MyKidCare.

5. HASIL KAJIAN

Pembangunan dan implimentasi adalah merupakan peringkat yang paling penting di dalam membangunkan sesebuah aplikasi. Bab ini akan membincangkan proses implementasi yang digunakan untuk membangunkan aplikasi berdasarkan rekabentuk dan spesifikasi sistem dalam bab-bab sebelumnya, antara proses – proses yang akan dibincangkan adalah berkaitan pembangunan antaramuka aplikasi, pembangunan pangkalan data dan implimentasi sistem. Ciri-ciri aplikasi yang telah dirancang dalam fasa sebelum ini akan direalisasikan untuk memenuhi keperluan pengguna.

Aplikasi ini dibangunkan dan memfokuskan kepada beberapa fungsi utama iaitu membuat carian dan tempahan pengasuh secara atas talian berdasarkan lokasi yang dipaparkan daripada pengesanan jarak lokasi google maps. Tiga pelaku(actor) yang terlibat adalah ibubapa(parent), pengasuh(sitter) dan juga admin yang diwujudkan secara backend untuk menguruskan pendaftaran pengasuh sebelum dapat dilihat oleh pengguna lain di dalam senarai pengasuh.

Rangka kerja bagi pembangunan aplikasi adalah menggunakan perisian Ionic versi 6.6.0 dan Node.js versi 12.12.1. Bahasa pengaturcaraan yang digunakan adalah TypeScript, Angular, php dan html. Manakala Kod editor yang digunakan untuk membangunkan aplikasi ini adalah Visual Studio Code(VSCode). Pelantar operasi Android digunakan untuk menjadi alat implementasi dan hanya peranti beroperasi Android boleh digunakan buat masa ini.

Sebanyak lima (5) table diwujudkan untuk menyimpan data pengguna iaitu user, book, fav, favparent dan juga availability. Rajah 4 di bawah menunjukkan senarai table yang diwujudkan dengan menggunakan mySQL.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(10)			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index Spatial More
2	name	varchar(255)	utf8_general_ci		No	None		Change Drop Primary Unique Index Spatial More
3	email	varchar(255)	utf8_general_ci		No	None		Change Drop Primary Unique Index Spatial More
4	password	varchar(255)	utf8_general_ci		No	None		Change Drop Primary Unique Index Spatial More
5	type	varchar(20)	utf8_general_ci		No	None		Change Drop Primary Unique Index Spatial More
6	userdesc	text	utf8_general_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
7	birthdate	varchar(20)	utf8_general_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
8	gender	int(1)			Yes	0		Change Drop Primary Unique Index Spatial More
9	firstaid	int(1)			Yes	0		Change Drop Primary Unique Index Spatial More
10	smokes	int(11)			Yes	0		Change Drop Primary Unique Index Spatial More
11	children	int(11)			Yes	0		Change Drop Primary Unique Index Spatial More
12	licence	int(1)			Yes	0		Change Drop Primary Unique Index Spatial More
13	car	int(1)			Yes	0		Change Drop Primary Unique Index Spatial More
14	location	varchar(255)	utf8_general_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
15	preferLocation	int(3)			Yes	0		Change Drop Primary Unique Index Spatial More
	Console xperience	varchar(50)	utf8_general_ci		Yes	0		Change Drop Primary Unique Index Spatial More

Rajah 4 Senarai table di dalam pangkalan data mykidcare

Untuk membolehkan aplikasi disambungkan dengan pangkalan data yang digunakan, link *URL API* yang digunakan akan ditambah ke dalam file *env.service.ts* seperti yang ditunjukkan seperti rajah 5 di bawah. Aplikasi akan membuat permintaan untuk mengambil atau menghantar data ke pangkalan data melalui API yang ditetapkan ini.

```

src > app > services > TS env.service.ts > EnvService
1 import { Injectable } from '@angular/core';
2 @Injectable({
3   providedIn: 'root'
4 })
5 export class EnvService {
6
7   //API_URL = 'http://lrgs.ftsm.ukm.my/users/ga03904/mykidcare_api/api/';
8   API_URL = 'http://localhost/mykidcare_api/api/';
9
10  constructor() { }
11
12

```

Rajah 5 File env.service.ts

Berikut adalah sebahagian daripada fungsi- fungsi yang terdapat di dalam aplikasi MyKidCare seperti di dalam rajah 6 dan rajah 7 fungsi seperti register(),login(), profile(),getprofile() dan lain- lain akan diambil daripada aplikasi melalui *api* ini dan data ditukarkan kepada bentuk JSON(JavaScript Object Notation) untuk membolehkan api dapat membaca data tersebut.

```

1 <?php if (!defined('BASEDIR')) {exit('<code>Forbidden Access</code>');}
2
3 class apiModel extends Model
4 {
5     function register($name,$email,$password,$type){
6         $sql = "insert into user (name, email, password, type, status) values ('$name','$email','$password','$type',1)";
7         return $this->exec_query($sql);
8     }
9     function login($email,$password,$type){
10        $sql = "select * from user where email='$email' and password='$password' and type = '$type'";
11        // echo $sql;
12        return $this->exec_query($sql,FETCH);
13    }
14    function profile($id,$name,$birthdate,$car,$desc,$experience,$firstaid,$gender,$licence,$location,$preferloc,$rate,$smokes,$children,$childreno,$lat,$lng){
15        $sql = "update user set name='$name',birthdate='$birthdate',car='$car',userdesc='$desc',experience='$experience',firstaid='$firstaid',gender='$gender',lic
16        // echo $sql;
17        return $this->exec_query($sql);
18    }
19    function profile2($id,$name,$desc,$location,$preferloc,$rate,$childreno,$lat,$lng){
20        $sql = "update user set name='$name',userdesc='$desc',location='$location',preferLocation='$preferloc',rate='$rate',childreno='$childreno',lat='$lat',ln
21        // echo $sql;
22        return $this->exec_query($sql);
23    }
24    function getProfile($id){
25        $sql = "select * from user where id='$id'";
26        return $this->exec_query($sql,FETCH);
27    }
28    function getParentAll(){
29        $sql = "select *, YEAR(CURRENT_TIMESTAMP) - YEAR(birthdate) - (RIGHT(CURRENT_TIMESTAMP, 5) < RIGHT(birthdate, 5)) as age from user where type='parent'";
30        return $this->exec_query($sql,FETCH);
31    }
32    function getBabysitterAll(){
33        $sql = "select *, YEAR(CURRENT_TIMESTAMP) - YEAR(birthdate) - (RIGHT(CURRENT_TIMESTAMP, 5) < RIGHT(birthdate, 5)) as age from user where type='babysitter'";
34        return $this->exec_query($sql,FETCH);
35    }
36 }

```

Rajah 6 File api_model.php.

```

25
26 function register()
27 {
28     $postdata = file_get_contents("php://input");
29     $req = json_decode($postdata);
30     // $id = $req->id;
31     $name = $req->name;
32     $email = $req->email;
33     $password = $req->password;
34     $type = $req->type;
35
36     $data = $this->apiModel->register($name,$email,$password,$type);
37     $data2 = $this->apiModel->createAvailability($email);
38     if(count($data) > 0){
39         $userObj->status = 1;
40         $userObj->message = "Registration successful!";
41     } else {
42         $userObj->status = 0;
43         $userObj->message = "Registration failed!";
44     }
45     $userObj->data = $data;
46     echo json_encode($userObj);
47 }
48
49
50 function login()
51 {
52     $postdata = file_get_contents("php://input");
53     $req = json_decode($postdata);
54     // $id = $req->id;
55     $email = $req->email;
56     $password = $req->password;
57     $type = $req->type;
58
59     $data = $this->apiModel->login($email,$password,$type);
60     if(count($data) > 0){
61         $userObj->status = 1;
62         $userObj->message = "Welcome!";
63     } else {
64

```

Rajah 7 File api.php.

Rajah 7 Kod di atas menunjukkan bagaimana *api* mengambil data objek yang dihantar daripada aplikasi di mana data tersebut akan di encodekan daripada bentuk objek kepada bentuk JSON dan berhubung dengan model api yang ditetapkan.

```

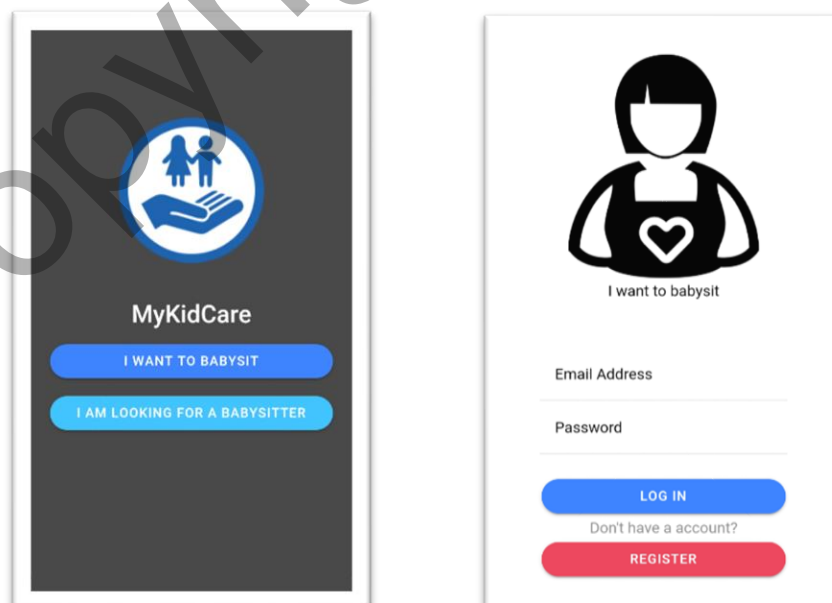
save(){
  var data = {user:this.user, availability: this.availability, geoLocation:
this.geoLocation};
  this.authSvc.postData(data,"profile").then(res=>{
    console.log(res);
    var obj:any = res;
    this.storage.set("userData", obj.data[0]);
    this.alertSvc.presentToast("Profile updated.")
  })
}

```

Rajah 8 File api.php

Kod diatas adalah merupakan kod pada aplikasi untuk mengemaskini profile berdasarkan data yang dimasukkan daripada pengguna dan memanggil *api* fungsi *profile()* yang terdapat di dalam *api*.

Aplikasi ini memerlukan pengguna untuk memasukkan email dan kata laluan yang sah bagi membolehkan pengguna masuk ke dalam aplikasi ini. Sekiranya pengguna masih belum mempunyai akaun yang berdaftar pengguna perlu mendaftar akaun terlebih dahulu. Rajah di bawah merupakan antaramuka utama untuk aplikasi Mykidcare. Terdapat 2 kategori utama pengguna utama yang membolehkan pengguna memilih mengikut berdasarkan kriteria yang diperlukan oleh pengguna.

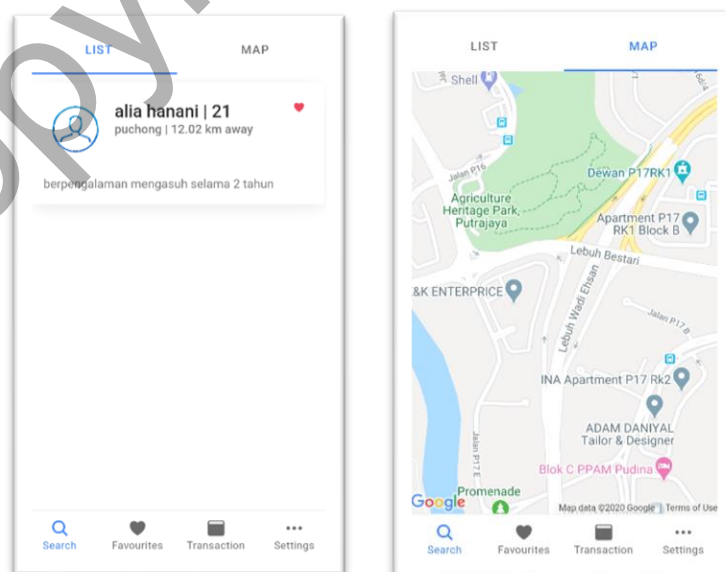


Rajah 9 Paparan skrin utama dan daftar masuk aplikasi MyKidCare

Untuk membuat pendaftaran pengguna perlu memasukkan maklumat yang diperlukan dan semua medan adalah mendatori untuk diisi bagi membolehkan pendaftaran dapat dilakukan. Paparan halaman bagi pendaftaran adalah seperti yang ditunjukkan pada rajah 10 di bawah.

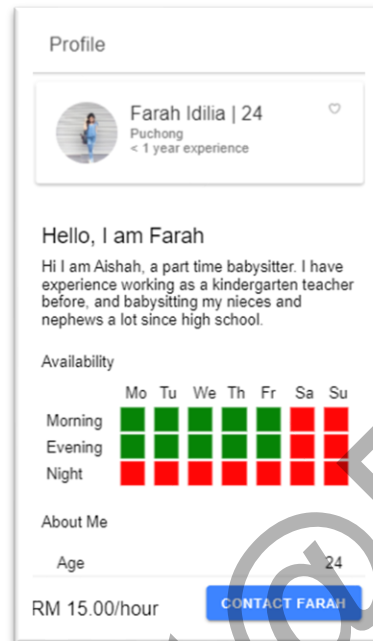
Rajah 10 Skrin halaman untuk pendaftaran ID pengguna

Apabila pengguna mendaftar masuk aplikasi, halaman ini akan memaparkan senarai pengasuh yang berdaftar sebagai laman utama. Pengguna boleh melihat senarai berdasarkan senarai yang dipaparkan ataupun melalui fungsi map yang menunjukkan lokasi pengasuh. Ini membolehkan ibubapa memilih lokasi yang sesuai dengan kehendak mereka.



Rajah 11 Skrin halaman perumah MyKidCare

Pengguna boleh melihat profile pengasuh dengan menekan pada nama pengasuh yang dikehendaki dan aplikasi akan memaparkan profile pengasuh yang mengandungi maklumat berkaitan nama, umur, *availability*, kadar caj upah perkhidmatan untuk satu jam, dan pelbagai maklumat lain.

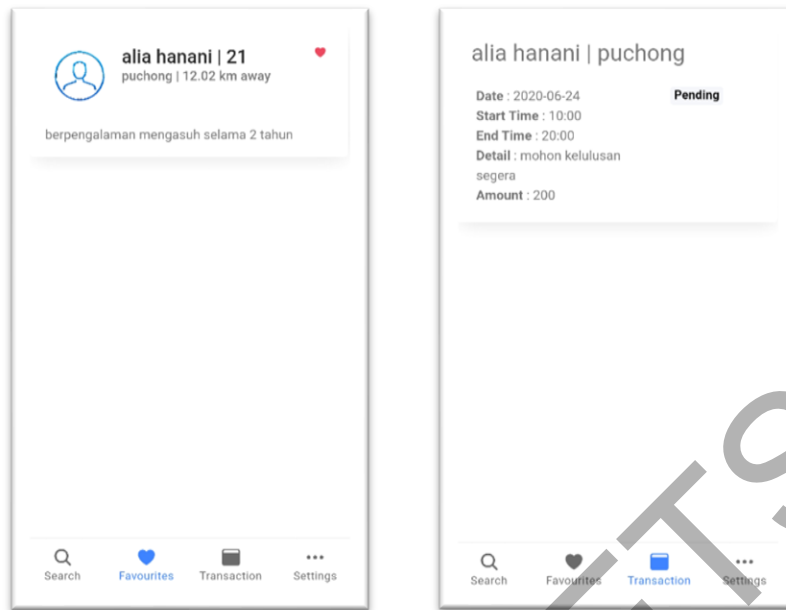
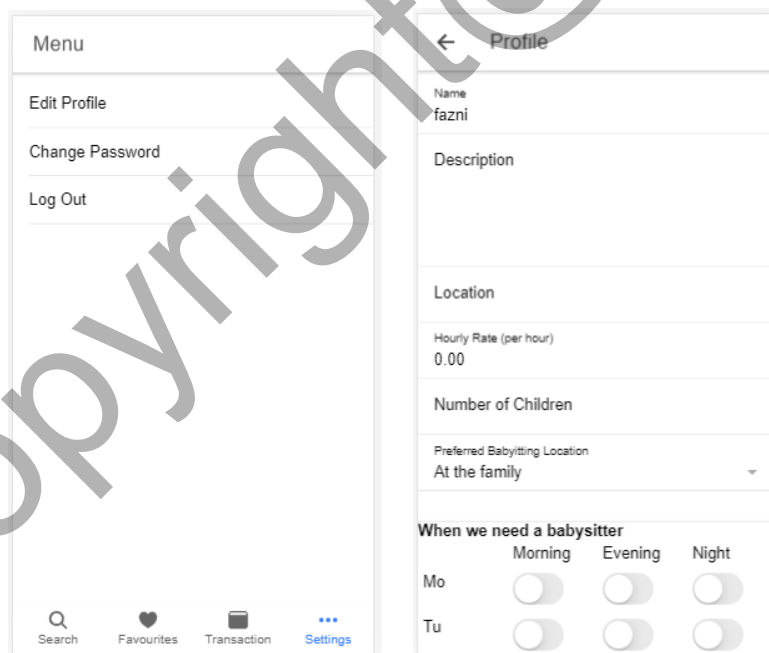


Rajah 12 Skrin halaman profile pengasuh MyKidCare

Terdapat 4 *tabs* menu utama di dalam aplikasi ini iaitu Search, Favourites, Transaction, dan settings. Berikut diterangkan secara ringkas penerangan berkenaan senarai menu *tabs* tersebut:

Tabs	Penerangan
Search	Search digunakan untuk memaparkan senarai pengasuh yang tersedia
Favourites	Untuk memaparkan senarai pengasuh yang telah di setkan favourite
Transaction	Untuk memaparkan semua transaksi dan sejarah transaksi
Settings	Untuk mengemaskini profile, menukar kata laluan dan juga daftar keluar daripada aplikasi

Jadual 1 Penerangan Tabs

Rajah 13 Skrin halaman *favourites* dan *transaction* penggunaRajah 14 Skrin halaman *setting* dan *edit profile* pengguna

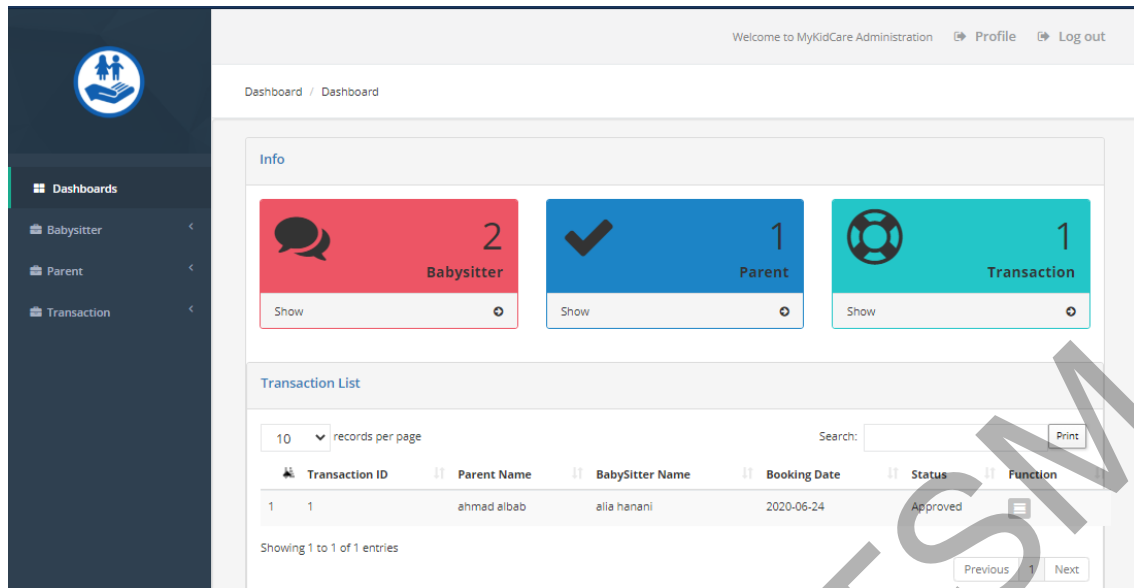
Halaman *favourites* membolehkan pengguna melihat senarai pengasuh yang telah ditetapkan sebagai pengasuh pilihan. Ini membolehkan pengguna membuat carian pantas melalui carian sebelum ini ataupun pengguna pernah menggunakan khidmat asuhan daripada pengasuh tersebut sebelum ini.

Halaman *transaction* membolehkan pengguna melihat status tempahan mereka ataupun melihat sejarah transaksi tempahan mereka. Senarai tempahan dan maklumat berkaitan tempahan akan dipaparkan seperti yang ditunjukkan pada rajah 13 seperti di atas.

Halaman update profile akan membolehkan pengguna mengemaskini profile mereka dan menetapkan masa kesediaan bagi pengguna jenis pengasuh untuk menerima tempahan khidmat asuhan seperti yang ditunjukkan seperti rajah 14 di atas.

Setiap pengguna yang mendaftar sebagai pengguna jenis pengasuh perlu melalui proses tapisan sebelum dapat menggunakan aplikasi ini. Ini merupakan salah satu prosidur keselamatan untuk mengelakkan daripada scammer dan sebagainya. Pentadbir akan memeriksa maklumat pengasuh sebelum meluluskan permohonan mereka.

Selain itu pengurusan pengguna dan juga transaksi juga amat penting untuk tujuan pemantauan dan penyelenggaraan data. Bagi tujuan itu satu *dashboard* khusus untuk menguruskan pengguna aplikasi MykidCare dibangunkan bagi memudahkan pentadbir melaksanakan tugas-tugas yang dinyatakan. Sistem pentadbir dapat dilayari menggunakan pelayar internet. Antaramuka pentadbir adalah seperti yang ditunjukkan pada rajah 15 di bawah.



Rajah 15 Skrin halaman *Dashboard* untuk kegunaan admin MyKidCare

Admin dapat melihat senarai pengguna iaitu jenis *Parent* dan juga *Sitter* serta melihat senarai transaksi pengguna aplikasi mudah alih.

Fasa implementasi ini mengambil masa yang agak panjang dalam mengenalpasti kod pengaturcaraan yang sesuai dan mengatasi masalah kod aturacara yang digunakan sesuai dengan kehendak aplikasi. Setiap kod diuji kebolehgunaan berkali – kali dan mengambil masa yang lama untuk disiapkan kerana dengan setiap ruangan memerlukan senarai data-data yang dicapai daripada pengkalan server serta kekangan ilmu pengetahuan. Seiring dengan fasa pembangunan, implikasi dan pengujian akan sentiasa berjalan sehingga proses pembangunan selesai sepenuhnya.

6. KESIMPULAN

Secara kesimpulannya aplikasi mudah alih MYKIDCARE ini akan dapat membantu ibu bapa untuk mendapatkan khidmat pengasuh dengan mudah cepat dan selamat. Selain itu aplikasi ini juga dapat membuka banyak peluang pekerjaan kepada masyarakat khususnya golongan yang berpendapatan rendah untuk menambahkan pendapatan. Mungkin pada suatu hari nanti aplikasi mudah alih MYKIDCARE ini akan menjadi satu aplikasi yang menjadi pilihan utama ibu bapa untuk dimuat turun dan digunakan.

7. RUJUKAN

Waterfall Software Development Model". 5 February 2014. Retrieved 11 August 2014.

Ina, M. 2018. Reka Bentuk & Teknologi : 1.1 Dunia Reka Bentuk. Retrieved November 22, 2019, from [https://www.slideshare.net/](https://www.slideshare.net/https://www.slideshare.net/MazIna1/reka-bentuk-dan-teknologi-85942282) URL : <https://www.slideshare.net/MazIna1/reka-bentuk-dan-teknologi-85942282>

Tanenbaum, Andrew S.; Steen, Maarten van (2002). Distributed systems: principles and paradigms. Upper Saddle River, NJ: Pearson Prentice Hall. ISBN 0-13-088893-1.

Nieh, Jason; Yang, S. Jae; Novik, Naomi (2000). "A Comparison of Thin-Client Computing Architectures". Academic Commons. doi:10.7916/D8Z329VF. Retrieved 28 November 2018.

Jennifer Hord, 2018. How Electronic Payment Works. URL : <http://communication.howstuffworks.com/electronic-payment2.htm>

Suhaimi Ibrahim et.all (1999) UTM Skudai, Kejuruteraan Perisian : Proses pengujian ms 177, 1999.

Arunkumar Khannur (2014) Partridge India, Structured Software Testing : The Discipline of software discovering Software Errors, Deriving Test Case and Scenarios from Use Case ms 294, 2014.