Project report appminton

Cplusco connect @2014

Cplusco Badminton adalah salah satu fasilitas yang disediakan oleh Cplusco untuk mereka yang ingin bermain badminton dengan fasilitas yang lebih mudah.

Cplusco Connect, developer team

Jl. Layur 04, Kakap Raya, Minomartani, Condong Catur   
Yogyakarta

Cplusco.com

Contents

[Summary of project 1](#_Toc405212004)

[Timeline Activity 6](#_Toc405212005)

[Masalah yang dihadapi 7](#_Toc405212006)

[Solusi masalah 8](#_Toc405212007)

[Software Information 9](#_Toc405212008)

[Company Information 9](#_Toc405212009)

# Summary of project

## Description of project

Project application yang sedang dikerjakan bernama Appminton. Yaitu aplikasi yang dibuat khusus untuk Cplusco Badminton, Brisbane yang dapat lebih memudahkan pengguna yang ingin bermain badminton. Aplikasi ini dapat digunakan untuk :

* **Scanner** yang digunakan untuk men-scan QrCode pemain saat akan bermain badminton pada session berjalan.
* **QrCode Card** adalah sistem yang digunakan untuk identifikasi pemain menggunakan **Scanner**.
* **Booking System**. Sistem booking untuk pemain setelah melakukan login.
* **Token/passes** adalah sistem yang digunakan untuk pemain dapat melakukan **booking** terlebih dahulu sebelum bermain.
* **Top Up** passes dapat dilakukan oleh pemain untuk menambah **passes** atau **token** dengan men-scan QrCode menggunakan Scanner.
* **RenewMembership** digunakan oleh pemain yang terdaftar sebagai member jika **passcard**/**Kartu anggota** yang digunakan sudah memasuki masa *expired* atau kadaluarsa.
* **Session List** adalah list dari session-session yang sudah berjalan, sedang berjalan dan session yang akan datang. Session list berisikan jumlah pemain yang berpartisipasi dan informasi keuangan.
* **Kartu anggota** yang digunakan untuk Top Up **passes**, bermain badminton, dan kartu keanggotaan pemain cplusco badminton.

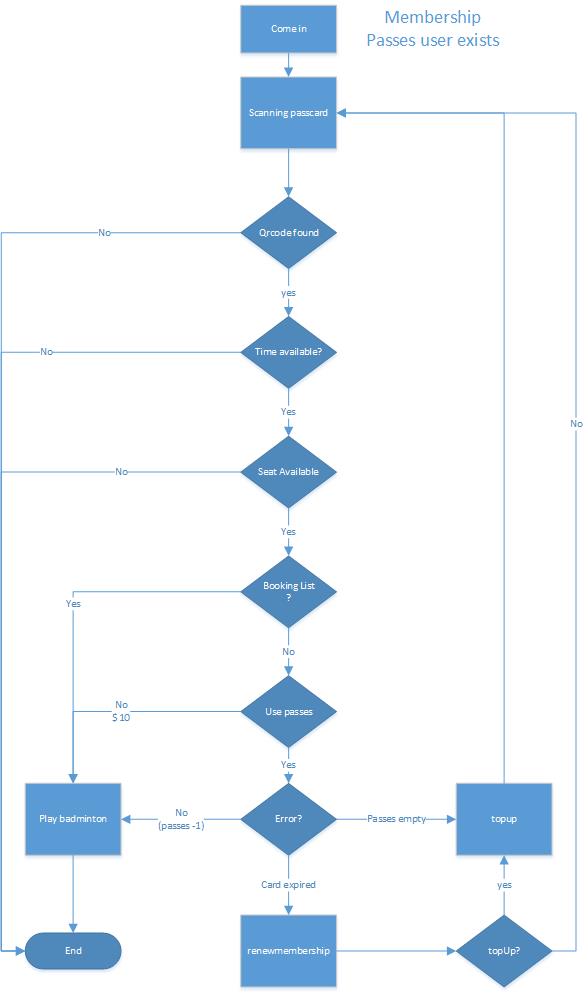
## Cashflow accounting

Aplikasi yang dibuat memiliki fitur penghitungan akuntansi yang masih terus di kembangkan untuk memudahkan dalam penghitungan uang masuk dan keluar. Serta mengembangkan informasi keuangan dengan bentuk chart sehingga pengguna mudah dalam memperkirakan naik turun arus keuangan.

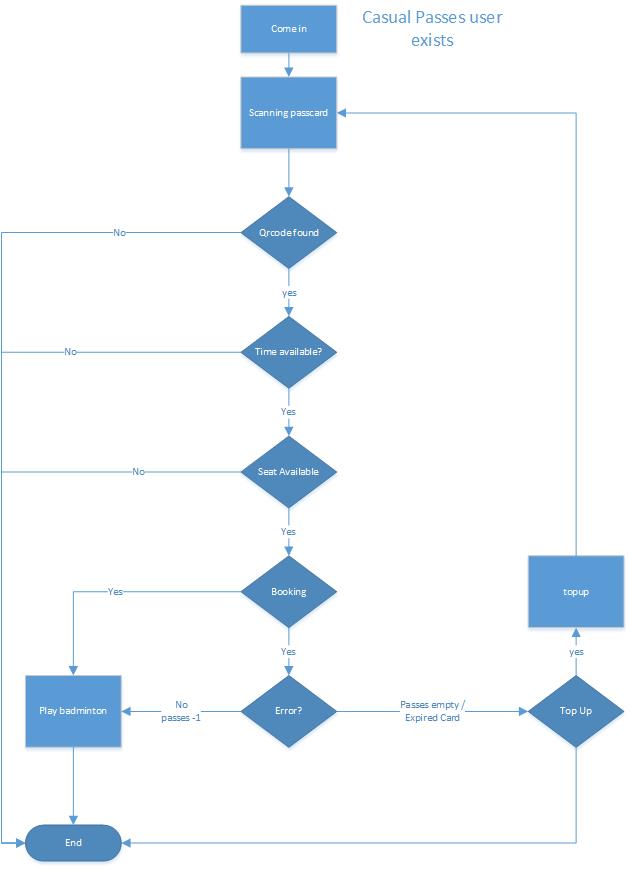
## EASy operating user interface

Appminton mengembangkan *user interface* yang mudah digunakan dan diadaptasi oleh pengguna. Baik pengguna desktop ataupun mobile.

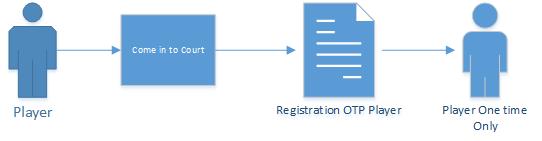
1. Membership Card



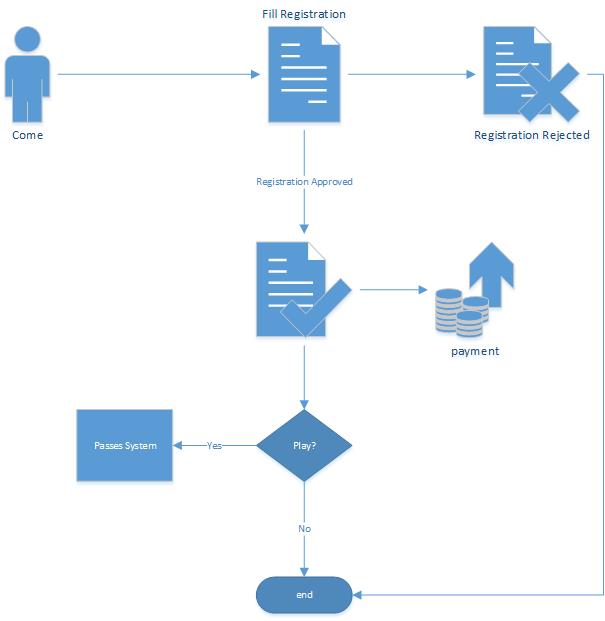
1. Casual Passes



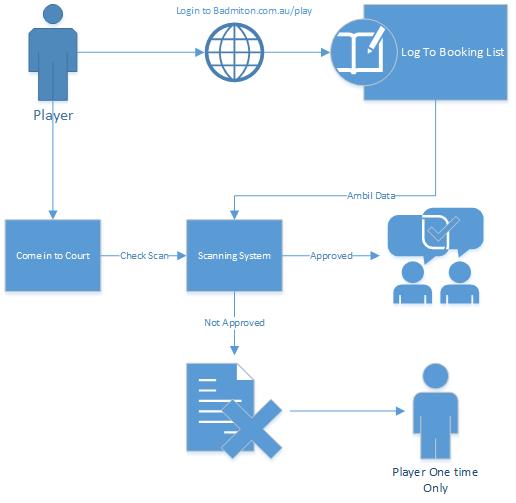
1. OTP (One Time Only)



1. Registration Flowchart



1. Booking Flowchart



# Timeline Activity

# Masalah dan Solusi

1. Masalah

* Kurangnya compatibility antara device dengan aplikasi yang akan digunakan. Seperti Scanner, photo web, dll.
* Adanya mis komunikasi dengan team saat pembuatan algoritma aplikasi.
* Kurang matang nya persiapan dalam peng-kode-an yang akan digunakan sehingga munculnya bug-bug program yang saling tumpang tindih dalam penggunaan hardcode.
* Membuat design aplikasi yang *user friendly* secara tampilan dan kemudahan penggunaan.

1. Solusi

* Penggunaan aplikasi tambahan bernama “**BridgeIt”** yang digunakan untuk menjembatani antara device dengan aplikasi yang akan digunakan. **BridgeIt** hanya dapat digunakan oleh device tertentu. Seperti **Apple** device, **Andorid** devices, dan **Windows Mobile** devices. Namun untuk beberapa device tersebut masih dalam tahap pengembangan oleh pihak **BridgeIt Developer**.
* Penyamaan pemahaman team dalam algoritma aplikasi sejalan dengan pembuatan **manual book** yang akhirnya team jadikan standard algoritma aplikasi.
* Persetujuan team mengubah code yang ditulis menggunakan Hardcode menjadi code yang lebih terstruktural dengan menggunakan Framework Object Oriented Programming.
* Meminta pertimbangan rekan kerja yang lain dalam design aplikasi dalam mengimplementasikan kemudahan penggunaan sistem **Appminton.**

# Suggestion

* 1. Pemahaman matang dalam pemilihan bentuk code, algoritma, dan perancangan aplikasi sangant diperlukan dalam persiapan pembuatan aplikasi. Beberapa langkah yang dapat ditempuh :
     + - 1. Perancangan alur program yang akan dibuat.
         2. Perancangan database yang akan digunakan.
         3. Perancangan algoritma yang akan diimplementasikan.
         4. Research tentang tools-tools dan plugin apa yang akan dipakai.
         5. Pengecheckan compatibility dengan device yang dibuat.
         6. Pemilihan bentuk code yang dibuat. Disini disarankan menggunakan framework agar mudah dalam identifikasi kesalahan dan pengembangan.
  2. Membuat jadwal task project agar lebih mudah dalam menentukan target waktu pembuatan.