BLUE BUS CAR SHARING APP

System Manual

Ali Aliyev, Yanwen Feng

April 26th 2016, Version 1.0

Blue Bus ATOS

Contents

Introduction

Blue Bus is a car sharing mobile app project for ATOS employees and potential customers. It enables to share cars during UK-wide trips from one ATOS site to another. This document is going to help programmers in further development know Blue Bus thoroughly.

Problem Statement & Challenge

Transport is major part of our everyday lives, especially for working people. Getting from one location to another in time sometimes can be very crucial. Gradual increase in number of cars, particularly in big cities results in traffic jams which is a nightmare for frequent transport users. Car sharing is a promising solution to this problem. There are already numerous companies that offer car sharing services around the world. In addition to that, there are opportunities for private car owners as well. Everyday drivers can offer rides to other people using various apps that are available in market. People can share their cars by offering free space to specified locations in specified times. This has multiple advantages like

- ✓ environment friendly - eliminating redundancy of cars for overlapping trips
- ✓ cost effective - more profitable use of car space
- ✓ corporate benefits - companies can save remarkably by reducing transport costs
- ✓ socialize - you get to know new people

We are required to make an app called Blue Bus which is meant to serve as car sharing app for ATOS employees and customers UK-wide. Car sharing is a relatively new service but because of its promising concept it is spreading fast. In order to start designing prototypes for the project we had to try out existing apps in the market so that we could have a basic notion how stuff works. Blue Bus is supposed to have all main functionalities that a car sharing app can offer. However it only allows searching and offering trips between ATOS sites inside UK and Northern Ireland.

Final Requirements

#	Final Requirements	Priority	Finished
1.	BBA shall schedule/arrange shared journeys to users with same destinations	1	Yes
2.	BBA shall be implemented on Android app platform	1	Yes
3.	BBA shall allow user to register with email(i.e. email cannot be the same as another user’ s), entering username, email, password, and age	1	Yes
4.	BBA shall have a database of records of user including email, password, name and age	1	Yes
5.	BBA shall allow user to log in with username and password	1	Yes
6.	BBA shall allow user to publish a ride	1	Yes
7.	BBA shall allow user to book a ride	1	Yes
8.	BBA shall allow user to choose the start point and destination on a map	2	Yes
9.	BBA shall allow user to type in a postcode or address and give the location	2	Yes
10.	BBA shall allow user offer return trips as well	2	Yes
11.	BBA shall allow user view and edit profiles	2	Yes

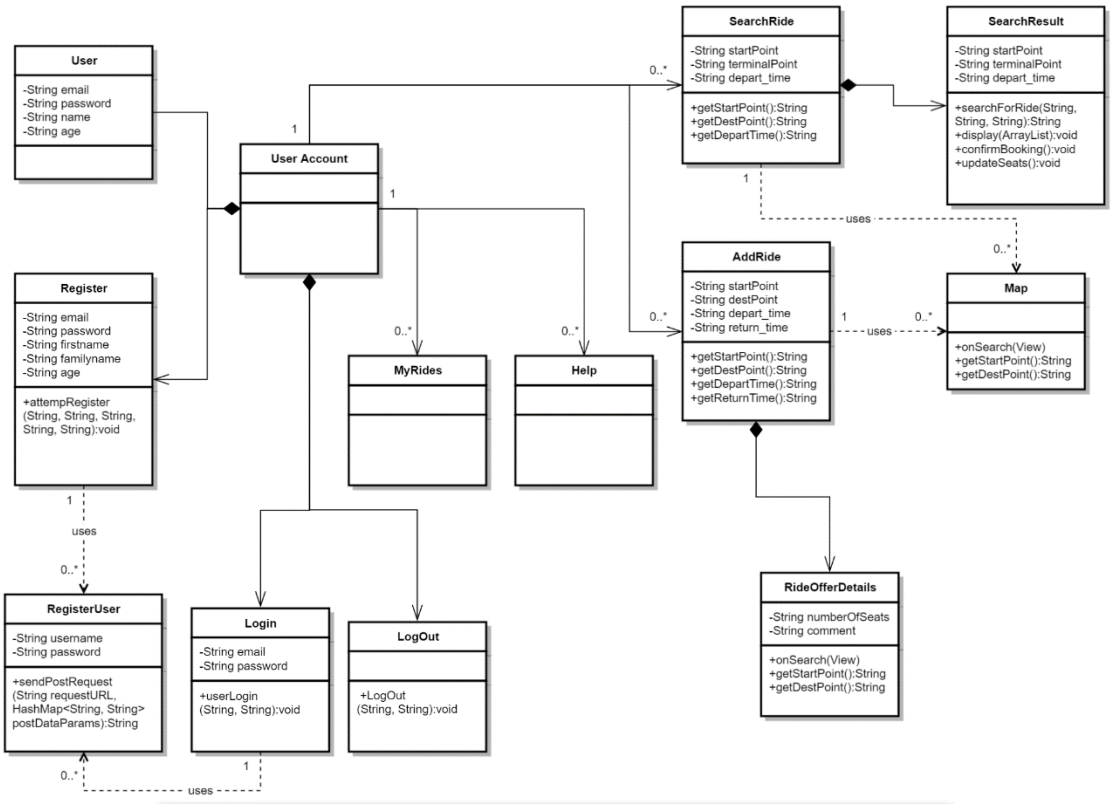
12.	BBA shall allow user to see his or her own previous rides	2	Yes
13.	BBA shall have the location of user(geolocation)	2	No
14.	BBA shall notify user if someone has booked his or her ride by email	3	Yes
15.	BBA shall be implemented on multiple platforms like Android, IOS and web browser(cross-platform app)	3	No
16.	BBA shall allow user to log in with Facebook or Twitter account	3	No
17.	BBA shall verify the phone number that user input when registration	3	No

Product Features

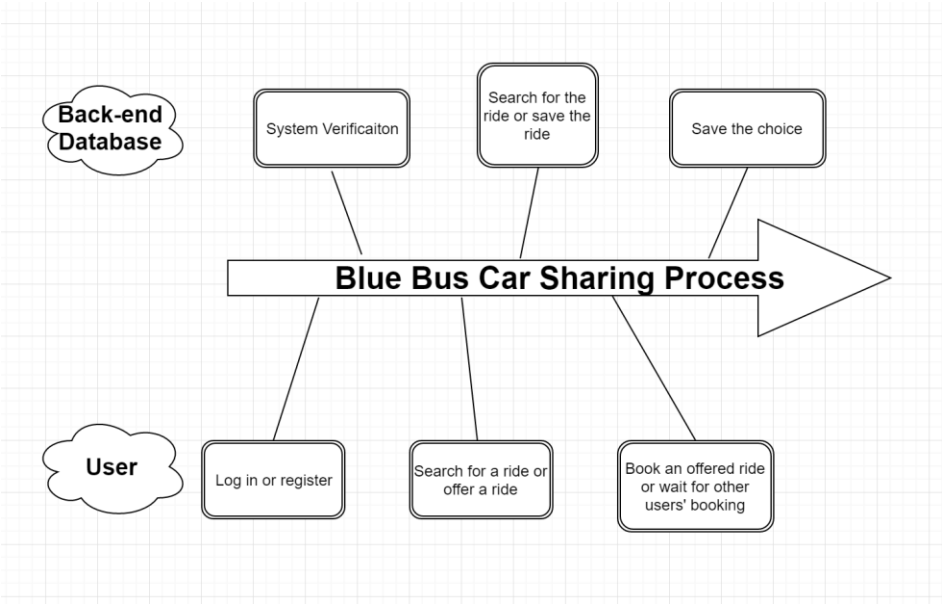
The following product features are prioritized based on how well they achieve business objectives:

- Blue Bus arranges shared journeys to users with same destinations
- Logo of Blue Bus looks cool and easy to recognise
- User has to log in before he or she offers a ride
- Blue Bus allows user to type in a postcode or address and give the location
- User can choose the start point and destination on a map
- Blue Bus has a database of records of user including email, password, name and age
- Blue Bus has a database of ride to record details of published(offered) rides

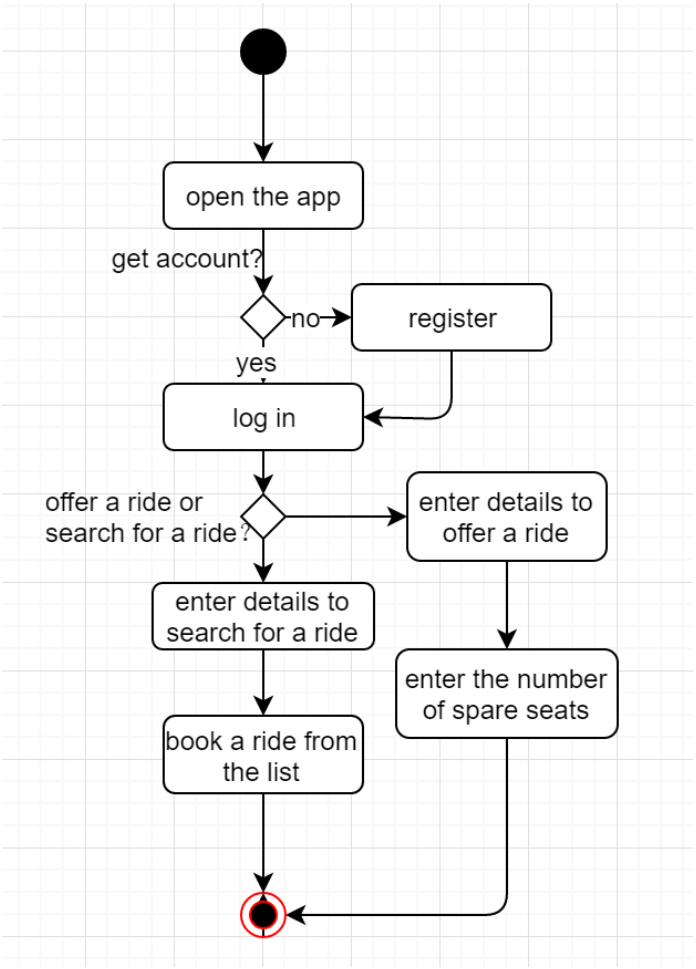
Architechtural Diagrams



Img1 Class Diagram



Img2 Process Diagram



Img3 Activity Diagram

Database&Automatic Email

We implement our database on phpmydamin and our server is on http://www.hostinger.co.uk/. We use PHP as the scripting language to establish connection with database. The following is the structure of two tables in our database.

Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Tracking

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change Drop Browse distinct values Primary More
2	gender	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
3	firstName	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
4	familyName	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
5	age	int(3)			No	None		Change Drop Browse distinct values Primary More
6	email	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
7	password	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More

Img4 table ‘users’

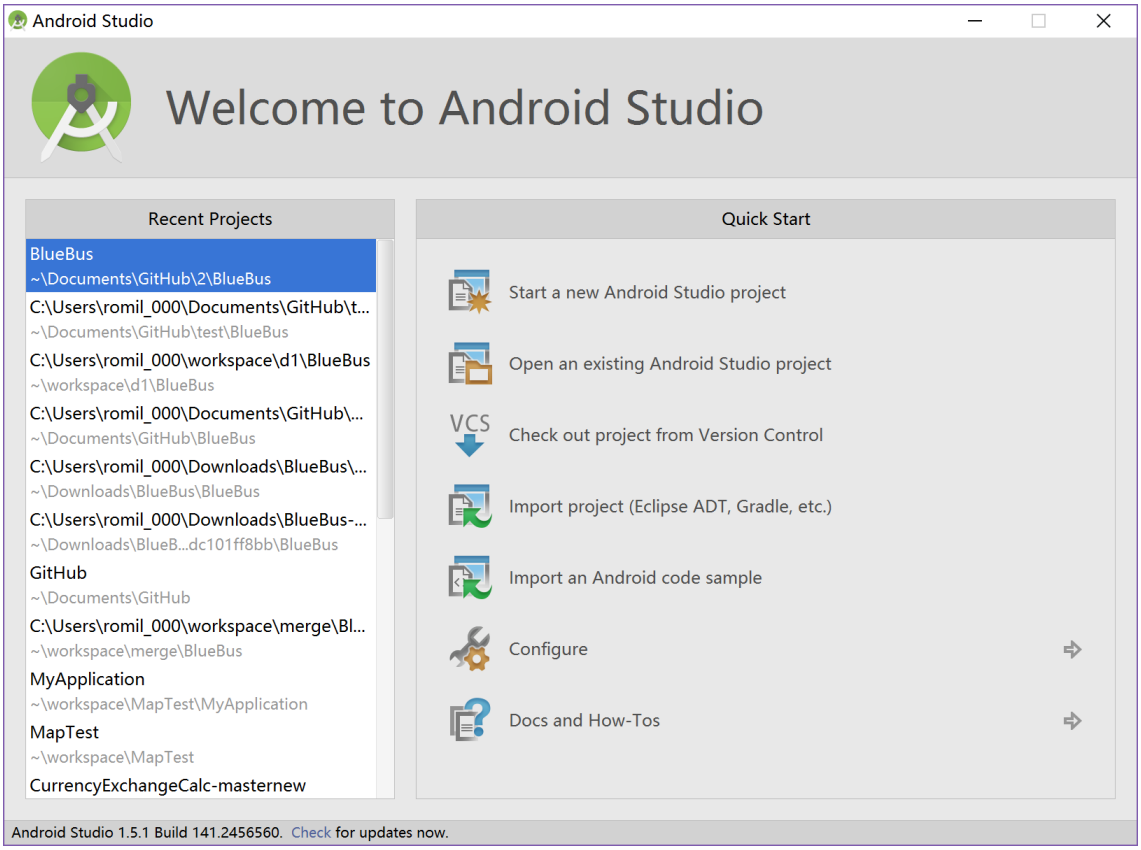
Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Tracking

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	start	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
2	destination	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
3	depart_time	datetime			No	None		Change Drop Browse distinct values Primary More
4	seats	int(2)			No	None		Change Drop Browse distinct values Primary More
5	comment	varchar(100)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
6	user	varchar(50)	utf8_unicode_ci		No	None		Change Drop Browse distinct values Primary More
7	id	int(11)			No	None	AUTO_INCREMENT	Change Drop Browse distinct values Primary More

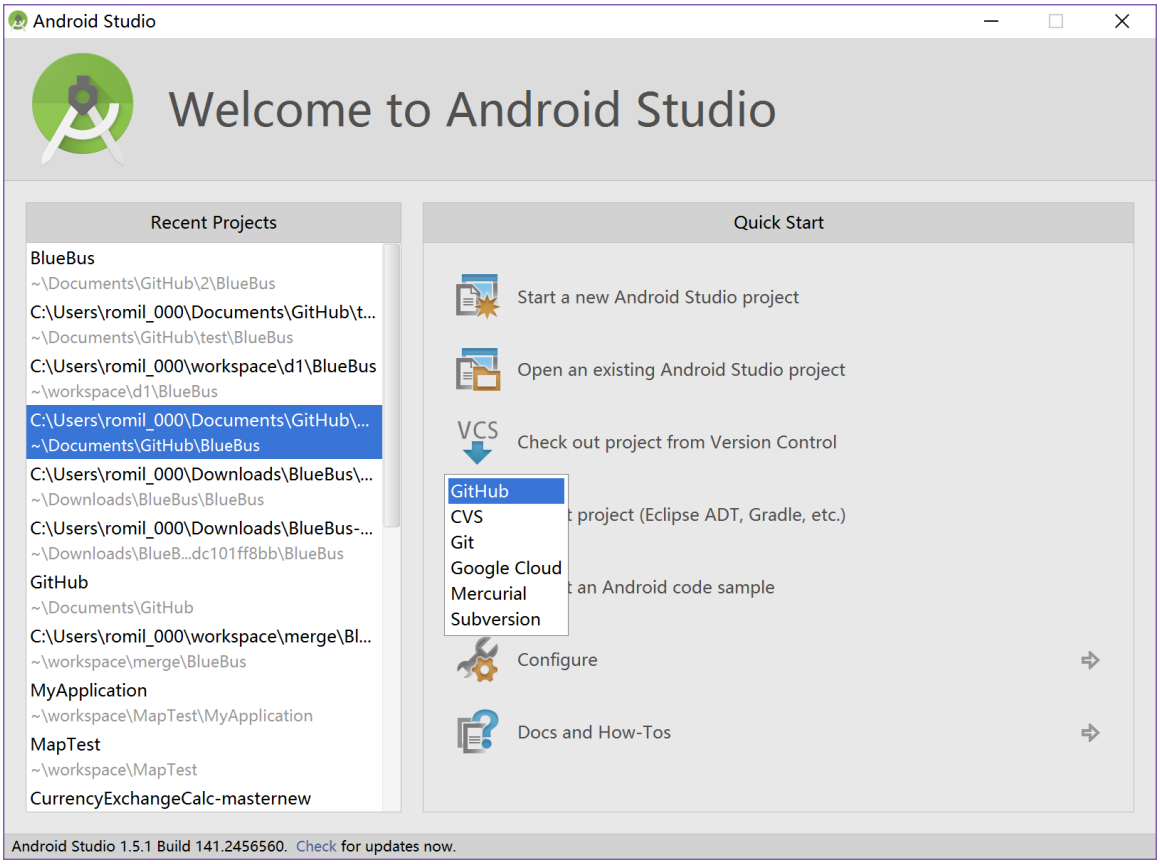
Img5 table 'rides'

Software Installation

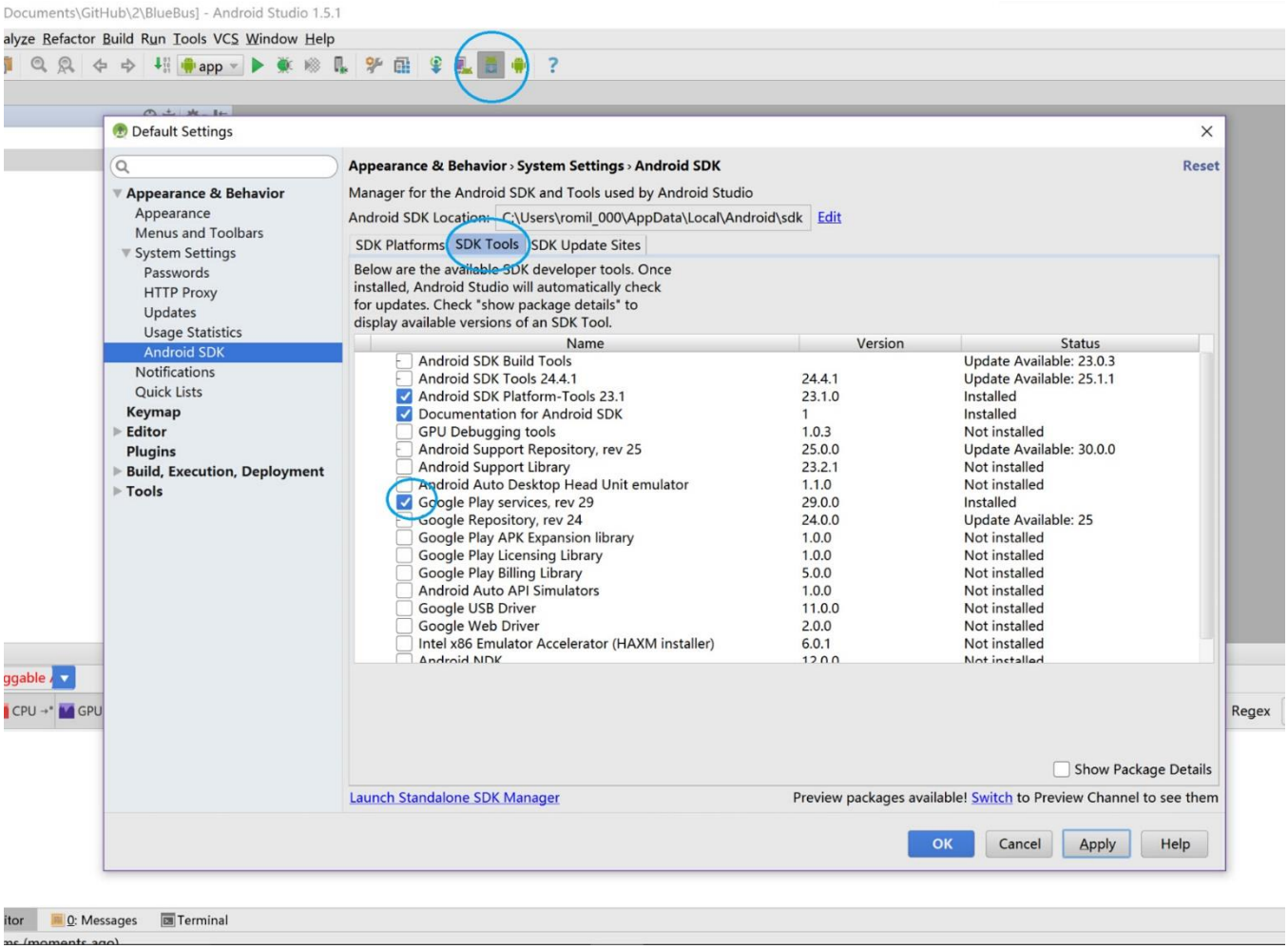
-Blue Bus is developed on Android Studio. Please download latest Android Studio.



-Choose Version Control to clone our project from
https://github.com/AliAliyev/BlueBus



-Click the icon of default setting as shown in the image. Turn to SDK Tools and install Google Play Services which is for Map page development.



-Now you can start coding for Blue Bus!

Using Blue Bus App

See USER MANUAL.PDF:
http://students.cs.ucl.ac.uk/2015/group6/UserManual.pdf

Testing & Evaluation

Purpose & Overview

This document is going to write about the actions that we take to test about Blue Bus App.
Blue Bus is a car sharing app on Android platform. Users can search for ride between ATOS offices on the app. To book a ride or offer a ride, users need to log in with username and password. Our app consists of several modules, registration and login, searching for a ride, offering a ride and my ride history.

in the Scope

In the scope of each module, we carry out functional test to make sure each class works as expected. The functions of UI components like navigation bar is easy to test, but tests on input fields are more complicated. For example, we need to do tests on the input field of the number of seats that a user wants to book. If we input letters, it makes user to try again as expected. When we input a number, it enables us to book a ride, even if the number of seats we enter is greater than available seats. The number is called invalid input. It indicates the method of validation check of the field needs improvement. After correcting it, we continue with boundary testing. It accepts the most number of available seats, so now we can roughly say the validation check of this input field is thorough. It means no unexpected input will be processed and no error will be invoked by it.

out of Scope

For tests out of scope, we focus on performance testing and integration testing. Performance testing tells the average response time and stability of the application. In our case, the app will not meet with large amount of input. The most amount of data that the app deals with is the result of searching a ride, which is limited to 20 records at most. This takes the app 1.8 seconds on average to send the query to the database, fetch the data from the database and load 20 records in the listview. Hence the respond speed of our application is excellent. Also, we carry out integration testing by simulating an actual user to use the app. We go through all the pages a user may turn to to test whether there are errors in connection of different components of the system and whether the application meets our final requirements.

Evaluation

There are four use cases that we wrote when we designed the product. We also go through these use cases. Fortunately, the result is what we expected and wrote in the documentation. In the end, our application meets all the must-have requirements and most of should-have requirements. During testing, we also discovered some problems that we had never thought of. For example, it will be good to set a ‘clear‘ button to make input more convenient. Due to the limit of time, we cannot perfect the application, but we can write them in the documents so programmers can refine them in further development.