1 ABS User Guide

1.1 Description

The ABS or Appointment Booking System is a Booking system designed for business use to enable clients, through an interactive GUI to view and select bookings for the given business. The Owner of the business can add employees, keep track of employee availabilities and thus schedule bookings appropriately.

2 Installation

2.1 Stable Builds:

The Appointment Booking System (ABS) requires java to be installed to run. If you need to install Java you can find it java.com. Download it and follow the instructions on the site to install.

To get the latest stable release for the ABS all one must do is download the latest version of the ABS.jar file.

Once Downloaded:

1. Open your command prompt or terminal
2. Navigate to the directory that contains the ABS.jar file
3. Run the command `java -jar ABS.jar` Or simple double click to run.
4. Enjoy the convenience of the Appointment Booking System!

2.2 Unstable Builds

2.2.1 WARNING

The next method may not work as it's meant to, if it works at all. It is recommended that you use the latest stable release of the ABS.jar file for a more pleasant experience BUT... if that didn't stop you, here's the instructions to build the system from the naked code.

If you like to live on the wild side and want the very latest from your Appointment Booking System, then you might want to download the repository and compile the program on your own**.**

To do so, you must have **ant** installed on your computer. Once downloaded you will have to install it, instruction of how to do so can be found here.

Once you have ant set up on your system, you are only a couple of commands away from having the very latest of the ABS.

1. Open your terminal or command prompt
2. Navigate to the directory which you have the downloaded ABS files (if they are zipped you will have to unzip them first)
3. Next step is to run `ant -f build.xml run`
4. Bask in the glory of the ABS & maybe read the usage notes if you need help

3 USER GUIDE

3.1.1 Register User

After launching the App select Register and enter your details accordingly. If you are a business owner select the tick box. If your details are valid and the email address unique your account will be created and you will be directed back to the welcome screen.

3.1.2 Logging in

From the welcome screen select Login and enter your details to be taken to the customer dashboard for customers or the owner dashboard for a business owner.

3.2 CUSTOMER FUNCTIONS

3.2.1 Making a Booking

From the customer Dashboard select the business you wish to make a booking for then select the Make a Booking option. This will display all the bookings for that business and if it is available it will show up green. Select the desired booking and you will be redirected to the dashboard.

3.2.2 Cancelling a Booking

From the Customer Dashboard select Remove a Booking to show all your current booking across all businesses. Select the Remove button next to the booking you wish to cancel. Upon success, you will be redirected to the dashboard.

3.2.3 View Business Info

From the Customer Dashboard select View Business Info to show the details about the business including their address and contact number.

3.3 OWNER FUNCTIONS

3.3.1 Registering a new Business

If you have logged in as an owner you can register a business via the business registration page. As an owner, you can register and manage multiple businesses.

3.3.2 Add Employee

After Logging in as an employee, click on the 'add employee' button. Then it's as easy as filling out the details as listed and then click add. Your new employee has been added. All that is left is to add an availability to them and they can be booked.

3.3.3 Add Availability

To add an availability to an employee, click on add availability button. Then write in the time and date into the box. Select the business and employee name and then finally click on add to add the booking for that employee.

3.3.4 Add A Booking for A Customer

In the case that a customer rings up and wants to book an appointment simply click on the book for a customer button, enter the customer's email address and then select the appropriate booking.

3.3.5 Deleting a Booking

To remove a booking from your business, click on the delete availability button. and simply click on the booking you wish to remove. And done, the booking will be removed.

4 SYSTEM INFO

4.1 DATABASE FILES

The utilities class can import/export data with default names/formats/locations or you can specify those things upon construction.

4.1.1 Default

By default, the data is stored in the data folder, located in the root folder. There are two text files, businessinfo.txt and customerinfo.txt. when the no argument utilities constructor is used it will look for those files in the data folder.

Default no-arg constructor:

```
/**
 *
 * Default constructor. Assumes businessinfo.txt and customerinfo.txt use
 * CSV and are stored in ./data/
 *
 */
public Utilities() {
        this.filePath = "./data/";
        this.businessInfoFileName = "businessinfo.txt";
        this.customerInfoFileName = "customerinfo.txt";
        this.splitChar = ",";
}
```

As an example, the customerinfo.txt is formatted bellow:

```
# Customer Info
# name,email,address,phone,password
Stacy,stacy.d@gmail.com,123 M Street bendigo vic 3550,0355555555,stacypassword
Jerry,bigj@live.com.au,45 Vic rd Melbourne Vic 3000,0414141414,bigjpass
```

4.1.2 Custom data files

The utilities class also supports specifying the location, filenames/types and list splitting character for the data.

By default, the printed lists are split back into arrays by separating the strings at a "," this is most commonly known as comma separated values (CSV), this constructor requires you to specify what character will be used for this purpose.

The custom data constructor:

```
/**
 * @param filePath
 *            Relative file path to data files.
 * @param customerInfoFileName
 *            Name for customer info file.
 * @param businessInfoFileName
 *            Name for business info file.
 * @param splitChar
 *            Character for splitting.
 */
public Utilities(String filePath, String customerInfoFileName, String
businessInfoFileName, String splitChar) {
        this.filePath = filePath;
        this.customerInfoFileName = customerInfoFileName;
        this.businessInfoFileName = businessInfoFileName;
        this.splitChar = splitChar;
}
```