CRC Cards

BookingSystem	
Responsibilities	**Collaborators**
Manage bookings	Bookings
Write bookings to CSV file	CSV
Update CSV file	

Booking	
Responsibilities	**Collaborators**
Add a booking	CSV
Update the bookings	Room
Remove booking	
Convert booking to CSV	
Convert CSV to booking	

Room	
Responsibilities	**Collaborators**
Store room data	

CSV	
Responsibilities	**Collaborators**
Read CSV files	CSVFileIterator

CSVFileIterator	
Responsibilities	**Collaborators**
Iterate through CSV files	

BookingFormSaveException	
Responsibilities	**Collaborators**
Generate error message	BookingSystem

Program	
Responsibilities	**Collaborators**
Run the program	BookingSystem
Create bookings	BookingFormSaveException
Manage bookings	
View analytics	

IO	
Responsibilities	**Collaborators**
Manage settings	Program

UI	
Responsibilities	**Collaborators**
Create a GUI for the program	AnalysisUI
	MakeBookingUI
	ManageBookingUI
	Login
	Program

Login	
Responsibilities	**Collaborators**
Login to the GUI program	Credentials

Credentials	
Responsibilities	**Collaborators**
Store credentials for the user	

AnalysisUI	
Responsibilities	**Collaborators**
GUI for analysis	Analysis
	UI
	Program

MakeBookingUI	
Responsibilities	**Collaborators**
Create a new booking through the GUI	UI
	Program

ManageBookingUI	
Responsibilities	**Collaborators**
Manage bookings	UI
Delete bookings	Program

SettingsUI	
Responsibilities	**Collaborators**
GUI for settings	UI
	Settings

Settings	
Responsibilities	**Collaborators**
Manage the theme of the GUI	UI
	Theme
	SettingsUI

Analytics	
Responsibilities	**Collaborators**
Generate analytics for the bookings	AnalyticsUI
	Program

TextBased	
Responsibilities	**Collaborators**
Provide a text-based system for the program	Program
	TextBasedDriver

TextBasedDriver	
Responsibilities	**Collaborators**
Run the text-based program	TextBased

UML Diagram

Analytics.java
+ getTotalProfit(ArrayList<Booking>): Integer
+ getProfitAverage(ArrayList<Booking>): Integer

Credentials.java
username: String
password: String

Login.java
mainUI: Scene

+ getUsers(): ArrayList<Credentials>
+ addUser(Credentails): void

AnalysisUI.java
analysisTable: TableView<Booking>
rooms: TableView<Room>
currentlySelectedCells: ArrayList<Booking>

VBox.java

Scene.java

MakeBookingUI.java
emailTxtField: TextField
nameTxtField: TextField
checkIn: DatePicker
checkOut: DatePicker
bookingTypeGroup: ToggleGroup
hotelType: ChoiceBox
roomType: ChoiceBox
occupancy: TextField
standardBooking: RadioButton
apBooking: RadioButton
rooms: Text
bookedRooms: ArrayList<Room>

UI.java
+ COMPANY_TITLE_FONT_SIZE: int
+ DEFAULT_SPACING: double
onThemeChange: Consumer<Theme>
onSave: OnSave<Booking>
contentPane: Pane
bookingSystem: BookingSystem

+ setOnSave(OnSave<Booking>): void
+ setOnThemeChange(Consumer<Theme>): void
+ getOnSave(): OnSave<Booking>
+ getOnThemeChange(): Consumer<Theme>
+ setBookingSystem(BookingSystem): void
+ getBookingSystem(): BookingSystem
setContentPane(Node): void

OnSave.java «interface»

ManageBookingUI.java
table: TableView<Booking>
rooms: TableView<Room>
currentlySelectedCells: ArrayList<Booking>

SettingsUI.java

Settings.java
+ theme: Theme

Theme.java «enumeration»

TextBased.java
+ scan: Scanner
- hotels: ArrayList<String>
noOfRoom: int
occupancy: int
roomType: String
hotelType: String
checkInDate: LocalDate
checkOutDate: LocalDate
ap: boolean

+ run(): void

TextBasedDriver.java
+ main(String[]): void

Booking.java
+ email: String
+ name: String
+ checkInDate: LocalDate
+ checkOutDate: LocalDate
+ isApPurchase: boolean
+ numberOfRooms: int
+ totalCost: int
rooms: ArrayList<Room>

+ getEmail(): String
+ getName(): String
+ getCheckInDate(): LocalDate
+ getCheckOutDate(): LocalDate
+ getIsApPurchase(): String
+ getNumberOfRooms(): int
+ getTotalCost(): int
+ addRoom(Room): void
+ toString(): String
+ toCSV(): CSV
+ fromCSV(CSV): void

BookingSystem.java
+ bookingIDCount: int
bookings: ArrayList<Booking>
bookingsArchive: ArrayList<Booking>
bookingFile: File
bookingArchive: File

+ readArrayIntoFile(File): ArrayList<Booking>
+ getBookings(): ArrayList<Booking>
+ removeBooking(Booking): void
+ addBooking(Booking): void
+ addAllBookings(Collection<? extends Booking>):
+ updateBookingsToFile(): void
+ updateArchiveToFile(): void
+ purgeOldBookingsFromArchive(): void
+ printBookings(): void

Program.java
+ MIN_APPLICATION_WIDTH: int
+ MIN_APPLICATION_HEIGHT: int
+ RESOURSES_PATH: String
+ SAVE_PATH_FILE: String
+ BOOKINGS_PATH: String
+ ARCHIVE_PATH: String
+ USERS_PATH: String
+ CSS_FOLDER_PATH: String
+ EMAIL_REGEX: String
+ formatter: DateTimeFormatter
+ rates: HashMap<String, int[]>
login: Login
ui: UI
bookingSystem: BookingSystem

+ Start(Stage): void
+ main(String[]): void
+ themeToFileName(Theme): String
+ onThemeChange(Theme): void
+ saveData(Booking): void
getRate(String, DayOfWeek): int

IO.java
+ loadSettings(): Settings
+ saveSettings(Settings): void
+ readFile(File): String

Room.java
+ hotelType: String
+ name: String
+ occupancy: int

+ getName(): String
+ getOccupancy(): int
+ toString(): String
+ toCSV(): String

Transaction.java
- resNum: String
- name: String
- checkIn: LoacalDate
- hotelTypes: String
- roomType: String
- ap: boolean
- totalCost: double

+ writeToCSV(): void
+ removeContent(): void
+ deleteFiles(File): boolean

BookingFormSaveException.java

Application.java

CSV.java
String: contents

+ iterator(): Iterator<String>
+ getSplitBy(String): ArrayList<C
+ push(): void
+ toString(): String

Iteratable<String>

- CSVFileIterator
- int: index

+ hasNext(): boolean
+ next(): String

Iterator<String>

Relationships labelled: Use, Extends