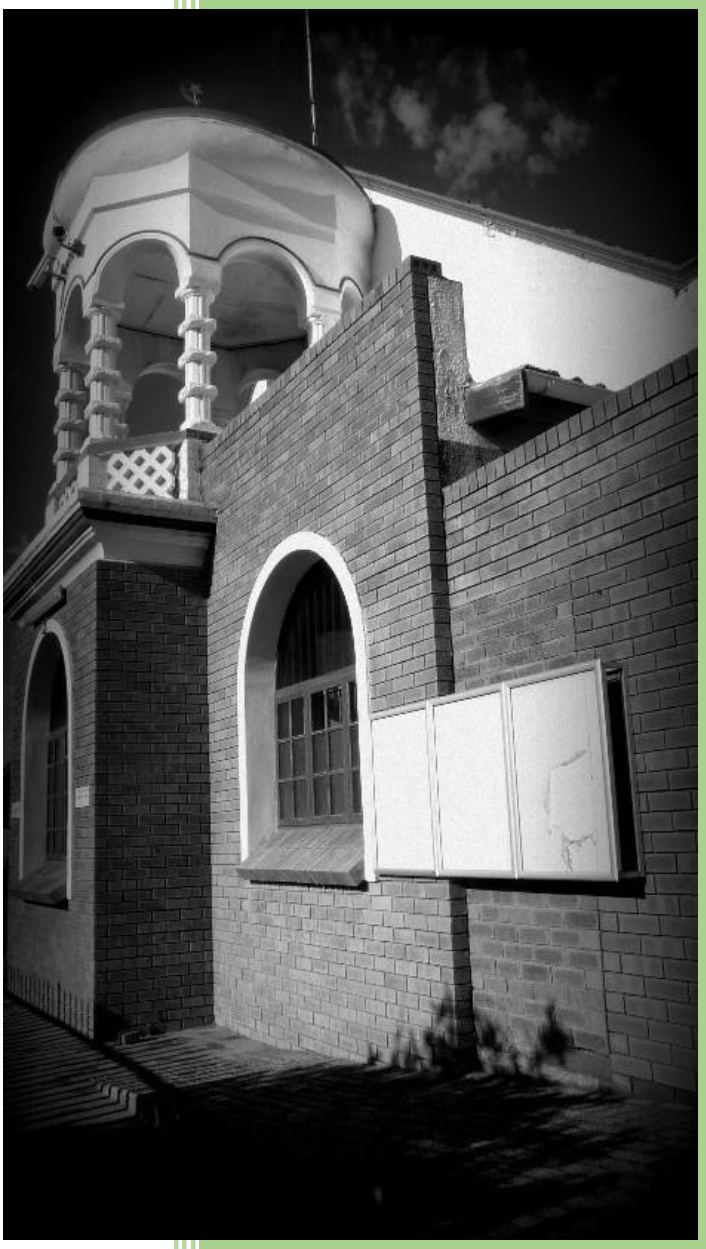
BRS GIS Ticket Management System



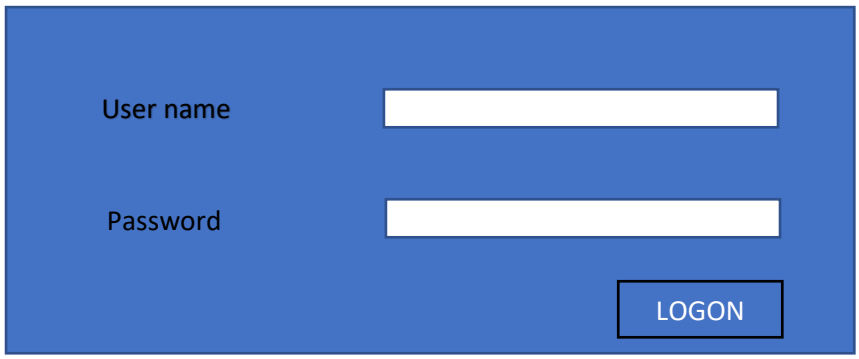
User Stories

- As a co-ordinator I want to capture a new fundraising event so that the resulting ticket sales can be tracked
- As an administrator I want to capture tickets allocated per person per event so that I can track payments received/outstanding
- As an administrator I want to capture payments received per person per event so that I can track payments received/outstanding
- As an administrator I want to capture tickets returned per person per event so that I can track un-sold tickets
- As an administrator I want to look up a person so that I can check their outstanding payments
- As an administrator I want to look up a person so that I can check their outstanding tickets
- As an administrator I want to query an event so that I can get a view of outstanding payments
- As an administrator I want to query an event so that I can get a view of un-sold tickets

User	Capture Event	Capture Ticket Allocation	Capture Returned Ticket	Capture Payments Received	Queries
Coordinator	yes	yes	yes	yes	yes
Administrator	no	yes	yes	yes	yes

LOGON

Screen



Logon Sample data (pre-populated)

Role

ID	RoleName
1	Coordinator
2	Administrator

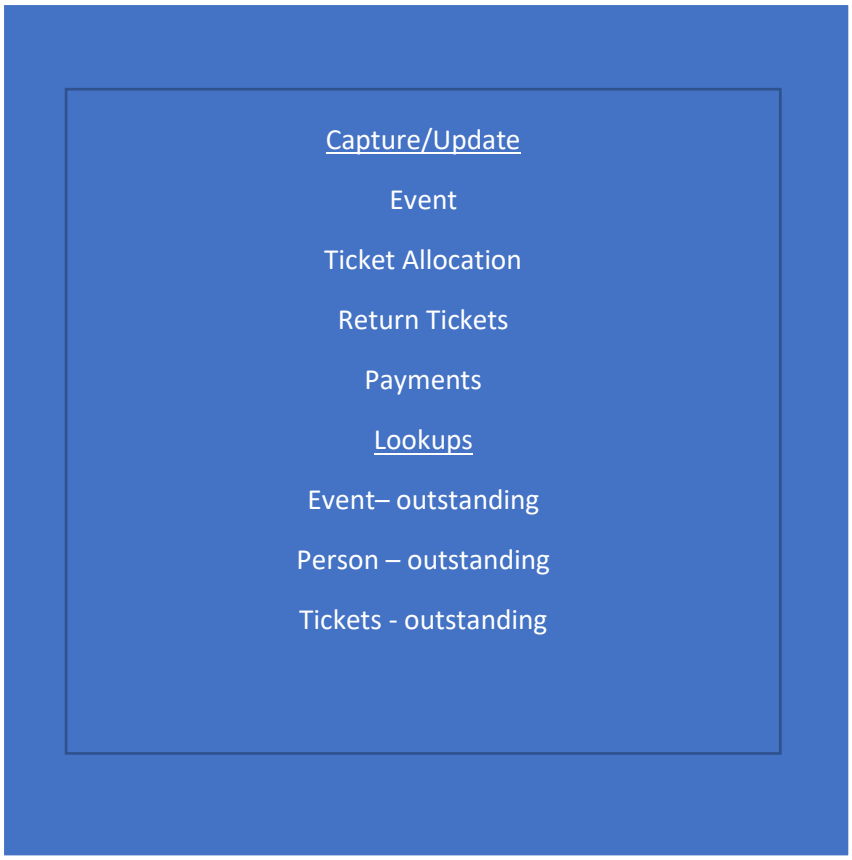
User

ID	Name	Surname	Email	username	password	RoleID
1	Joe	Soap	js@gmail.com	jsoap	4s45s	1
2	Amina	Frieslaar	af@gmail.com	afrieslaar	E4e55	2
3	Warrel	Jones	wj@gmail.com	jwarrel	Ddaw1	2
4	Marrel	Appleby	ma@gmail.com	mappleby	Ff55dv	2
5	Jainap	Reeve	jr@gmail.com	jreeve	Dsvsd2	2

Note:

- Only users with User record can log on

MENU

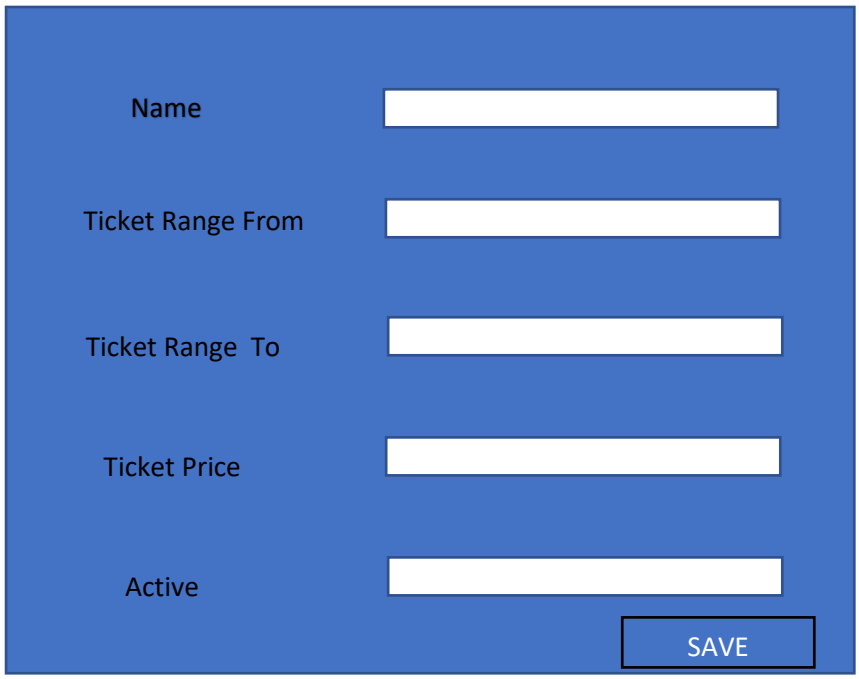


Note:

- Only User where RoleID = 1 can see Event menu item
- User where RoleID = 1 or where RoleID = 2 can see all other menu items

Event

Screen



Sample data

Event

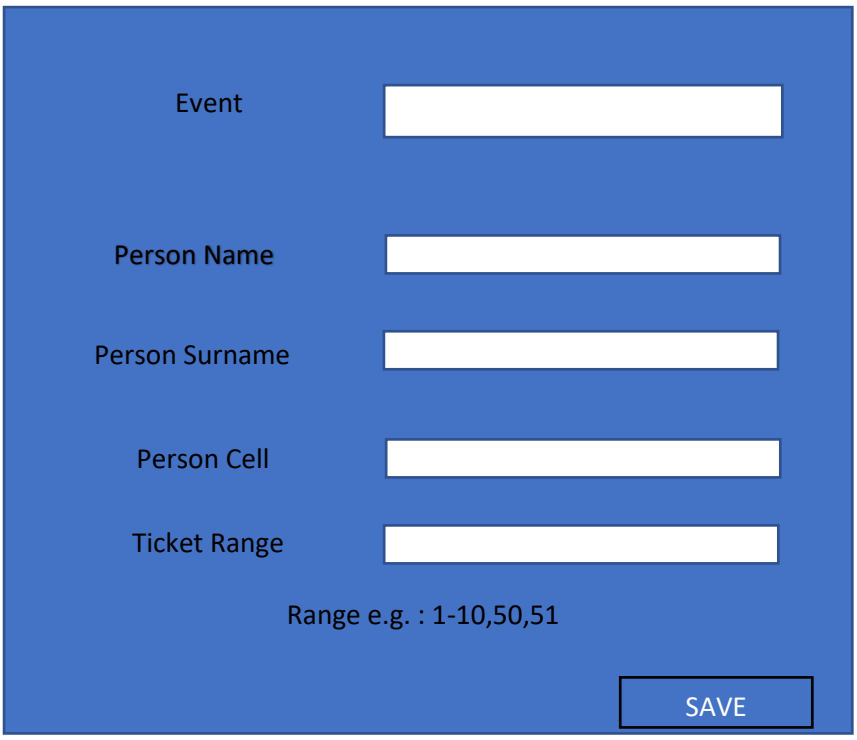
ID	Name	TicketFrom	TicketTo	TicketPrice	Active
1	FoodFairApril2018	1	500	100	false
2	FoodFairSept2018	1	500	120	true

Note:

- Only User where RoleID = 1 can capture Event records
- Cannot update Event when Active = false
- If the Name exists in Event table then populate screen with the data

Ticket Allocation

Screen



Sample data

Person

ID	Name	Surname	ContactNumber
1	Salie	Deen	082 501 2456
2	Rhoda	Scheepers	083 985 2244

TicketAllocation

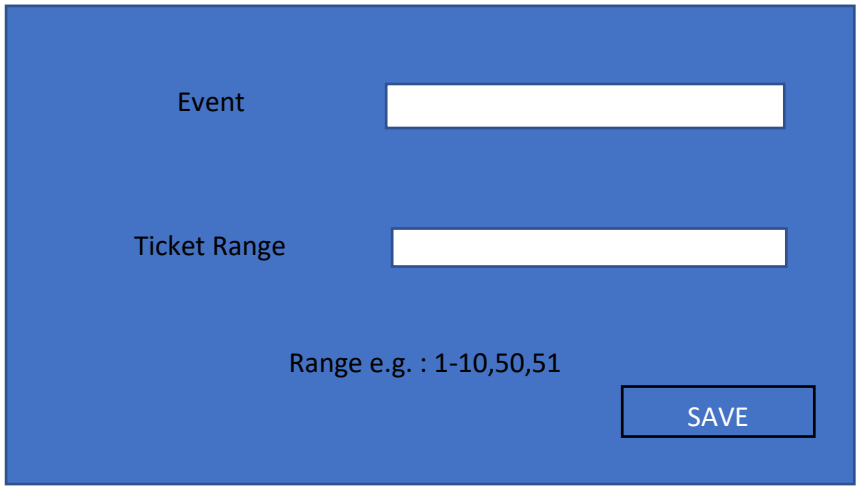
ID	TicketNumber	Paid	EventID	PersonID
1	1	true	2	1
2	21	false	2	2

Note:

- Event field validates against Event table
- Cannot allocate tickets Event Active = false
- Cannot Allocate Tickets where Event does not exist
- Ticket Range e.g. 1-10,50,51 will insert 12 records in TicketAllocation table for Person and Event

Return Tickets

Screen

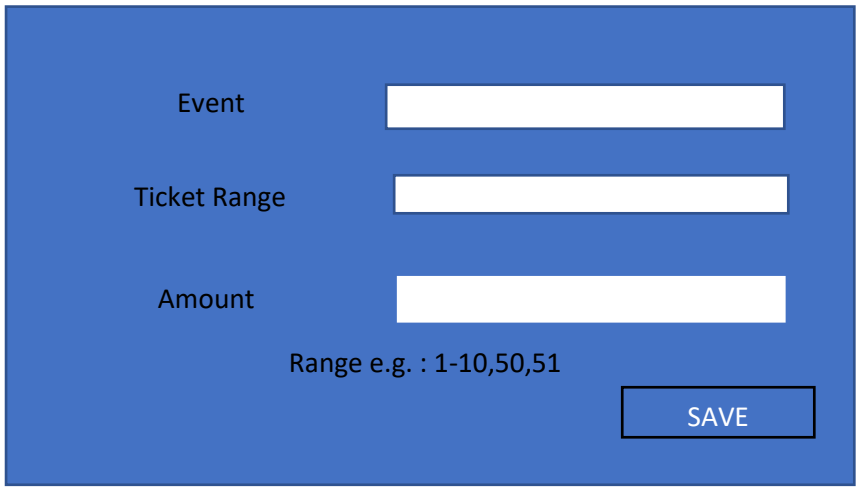


Note:

- Event field validates against Event table
- Cannot return tickets Event Active = false
- Cannot return Tickets where Event does not exist
- Ticket Range e.g. 1-10,50,51 will delete 12 records in TicketAllocation table for Person and Event
- If Person record has no other TicketAllocation records linked to it, then delete the Person record

Payments

Screen



Note:

- Event field validates against Event table
- Cannot do payments for tickets Event Active = false
- Cannot do payments for tickets where Event does not exist
- Amount must correspond to Ticket Range e.g. 1-10,50,51 will update 12 records in TicketAllocation table for Person and Event

Lookups

Event – outstanding

Capture Event name and Displays:

- payment amount outstanding vs total payment amount for Event
- tickets outstanding vs total tickets for Event

Person – outstanding

Capture Person and Displays:

- payment amount outstanding per Person
- unpaid tickets per Person

Ticket – outstanding

Capture Event name Displays:

- names of Persons who owe payments
- names of Persons listing unpaid tickets

Webservice

Exposed end-points functionality

- Insert Event
- Update Event
- Insert Person and TicketAllocation
- Update Ticket Allocation
- Delete TicketAllocation
- Delete Person
- List Events
- List for Lookups menu items

Non-Functional Requirements

- Android application
- Hosted mysql database
- Database updated in real time
- Back end persistence framework (e.g. hibernate) for CRUD code
- Back end exposed as webservice (e.g. using jersey)
- Apk to be manually distributed