

A

Project Report

On

# **Visitor Management System for Raj Bhavan, Maharashtra (Mobile App Development)**

AT

National Informatics Centre, Maharashtra State Centre, Mumbai  
(Ministry of Communications and Information Technology)

Submitted to:

Pimpri Chinchwad college of engineering, Pune

In Partial Fulfilment of the Requirement

for the Award of the Degree of

## **Master of Computer Application(MCA)**

Designed and Developed By:

**Shalinikumari Prakashbhai Mane**

## **Declaration**

I hereby declare that the work being presented in the project entitled "**Visitor Management System for Raj Bhavan, Maharashtra (Mobile App Development)**" for the requirement of the award of **Master of Computer Application(MCA)** is an authentic record of my work carried out for the period from 01st June 2022 to 30th November 2022 under the guidance of **Mrs. Ireni Akoijam Scientist 'F'** and **Mr. Amodkumar Pradip Suryawanshi, Scientist 'B' NIC Mumbai.**

**Shalinikumari Mane  
(MCA 4<sup>th</sup> Semester)**

## **ACKNOWLEDGEMENT**

It is a great sense of satisfaction and a matter of privilege for me to have an opportunity to work at **National Informatics Centre, Mumbai**. I wish to express my heartiest thanks, sense of gratitude, and sincere regards to everybody who helped me and guided me at all stages for the successful completion of this project.

It is my pleasure to thank specially to **Mr K. P. Pariselvan, Scientist 'G' and State Informatics Officer, NIC Mumbai**, to whom I owe a lot for giving me an opportunity to undergo my training in this organization and enhance my skill.

My Profound thanks and gratitude to **Mrs. Ireni Akojam Scientist 'F'**, **Mr. M.A. Qayum, Scientist 'F'**, and **Mr. Amodkumar Suryawanshi, Scientist 'B' NIC Mumbai** for their support, guidance and monitoring me at every stage. They always paid enough attention and gave careful thought to the requirements which clarified a number of otherwise obscure points and sharpened my vision on this project.

It gives me great pleasure to express my gratitude to **Mr. M.A. Qayum, Scientist 'F', NIC Mumbai** for his support, guidance and monitoring me at every stage. He always paid enough attention and gave careful thought to the requirements which clarified a number of otherwise obscure points and sharpened our vision on this project.

I would like to express my sincere regards to **Dr. G. N. Kulkarni, Director, Pimpri Chinchwad College of Engineering, Pune**, **Prof. Anjana Arakerimath Head of Department MCA**, **Mr. Rajkamal Sangole, Internal Project guide** for their valuable support

I am grateful to all other staff members of NIC, Maharashtra State Centre Mumbai group for providing me support and help as when required. The project would not reach in its present form without their inspiration and encouragement, which kept my spirits high during the project.

Last but not the least I would like to express my thanks to my parents for their moral and financial support without which I could not have been at the light of the day. They have done everything possible to help me to reach at this position.

*Shalinikumari Mane*

## **PREFACE**

The basic objective of providing computer education is to develop a scientific and logical approach, promote an understanding of basic tools and concepts of programming, and to enhance analytical capabilities.

Is this all possible by limiting students only to books and classrooms? The answer is 'NO'. The exposure of students to the real outside world is equally necessary where the skills are to be tested ultimately. The live project, infects the sharpness of the skills, expands the bookish knowledge removes the rigidity in approaches, and helps in being flexible.

As per the schedule of the MCA curriculum, I was given an opportunity to understand the practical aspect of Software Development. I have taken this golden opportunity to build a new real-time system. It is a great experience that has given me the feeling of the actual environment of Application Development.

Live projects are different from normal projects. Every time you build up a module you do have not to think of its aim but also keep in mind that it should work in the coming years, without any problem. One should carefully identify the limitations and try to reduce them to a minimum.

The application also provides security, strict checking constraints, and flexibility as per need so that system can run in the future for a long time, without any problem.

***ShaliniKumari Mane***

## Contents

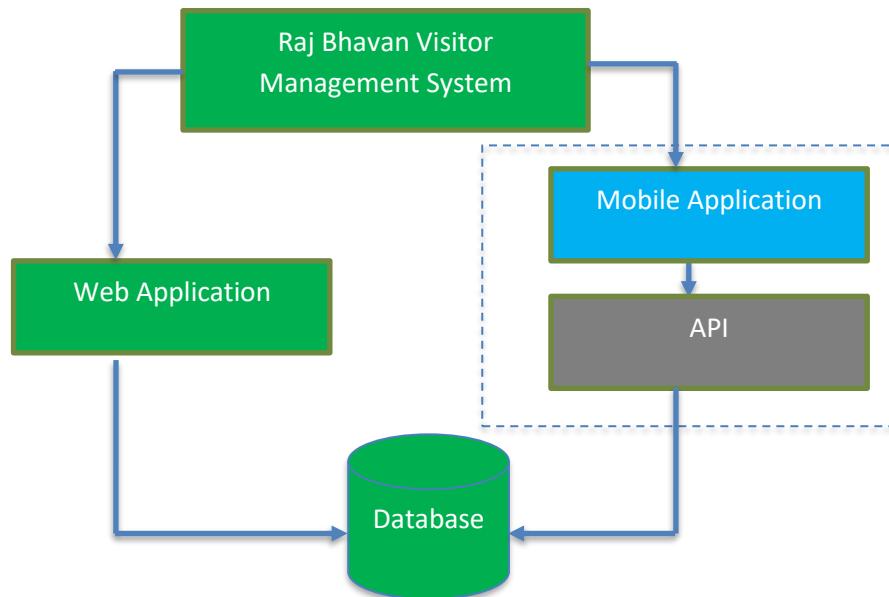
<b>Section</b>	<b>Title</b>	<b>Page No.</b>
1	Introduction	6
1.1	Introduction to Organisation	8
1.2	Technology Stack	10
1.3	Introduction to Software	11
2	Software Requirement Specification	13
3	Software Design Document	28
4	User Manual	47
5	Conclusion	62
6	Future Scope of the Project	62
7	Experience Gained	63
8	Appendix	64
9	Bibliography	82

# INTRODUCTION

Raj Bhavan, Mumbai, Maharashtra is the official residence of the Governor of Maharashtra and it is one of the most beautiful buildings in Mumbai City. It is having beautiful and naturally significant features as it breathes a history of century and a half. Its rich history and precious collections of furniture and doors attracts many citizens to visit it. The visitors of the Raj Bhavan often visit the residence to find answers for their curious minds about history and the items kept here.

These lovers of art and history have to book their tickets based on the slot-system before visiting Raj Bhavan. They have to book the Sunrise Gallery and Bunker visit before visiting so that they can get a smooth access into the Raj Bhavan. Earlier the system was completely based on a web application alone i.e., the user had to book his/her tickets only via a browser, using the official Raj Bhavan website. However, this project aimed to build a mobile application which is becoming more convenient for citizens as they can book tickets using their smart phone easily. In today's world, everyone owns a smart phone, due to which people prefer using mobile application versions of a software over the web applications, which needs use of a desktop or a laptop. Hence, there raised a need for a mobile application for android and iOS users.

The project included two parts – A. Development of API for fetching data from the database and sending the same to the mobile application. B. Development of Mobile application. The part A of API development was designed and developed by other member and the part B of mobile application development was taken care by my other team member.



## **1.1 Overview of the Organisation**

### **NATIONAL INFORMATICS CENTRE (NIC)**

National Informatics Centre (NIC) is a premiere S&T institution of the Government of India, established in 1976, for providing e-Government/e-Governance Solutions adopting best practices, integrated services and global solutions in Government Sector.

In 1975, the Government of India strategically decided to take effective steps for the development of information systems and utilization of information resources and also for introducing computer-based decision support system (informatics-led development) in government ministries and departments to facilitate planning and program implementation to further the growth of economic and social development. Following this, the Central Government nucleated a high priority plan project "National Informatics Centre (NIC)" in 1976.

NIC has played an important role of an active catalyst and facilitator in informatics development program in Governments at the national, state and district levels, which made them take policy decisions to create knowledge societies – societies that can exploit knowledge to derive competitive advantage using the opportunities provided by digital technology. NIC has been instrumental in adopting Information and Communication Technology to reach out across India i.e., by implementing IT applications in Social and Public Administrations. NIC has spearheaded the e-Governance drive in the country for the last three decades building a strong foundation for better and more transparent governance and assisting the governments endeavor to reach the unreached.

NIC Maharashtra State Centre, Mumbai is located at New Administrative Building, Mantralaya. NIC Maharashtra State Centre provides software development, internet, communication services and Video conferencing via NICNET to State and Central Govt. departments in the state of Maharashtra. Besides the State Centre, NIC offices are also located in the High Court & its Benches, Central Government offices and 36 district centers located at respective Collectorate offices. These centers provide informatics support to Collectorate offices, Zilla Parishad, District courts and all allied offices of State and Central Government.

NIC has implemented many "network centric" applications for program implementation in various ministries and departments, using state-of-the-art software tools. During the 1980s and early 1990s, the policy thrust was on creating a "decision support system" for development planning and responsive administration in governments. NIC has vast expertise and experience in the design, development and operationalization of various e-Government projects in the areas of Public Administration and Governance, such as Agriculture & Food, Animal Husbandry, Fisheries, Forestry & Environment, Industry, Health, Education, Budget and Treasury, Fiscal Resources, Transport, Water Resources, Court Management, Rural Development, Land Records and Property registration, Culture & Tourism, Import & Exports facilitation, Social Welfare Services, Micro-level Planning etc.

Website of NIC: [www.nic.in](http://www.nic.in)

**PRINCIPAL OBJECTIVES OF NIC:**

Development of Computerized Information Systems in various sectors of the economy for planning, project, monitoring and decision support.

Promotion of Information Culture at District, State and national level.

Improvement of analytical capability and presentation of Information utilized for Central, state and District level planning.

Development of modelling, forecasting and simulation techniques that are required for Planning and decision support.

Establishment of a computer communication networks for ensuring ready access of information across the nation enabling the optimal use of resources.

Evolving standards for data collection, compilation and dissemination with the cooperation of related agencies at District, State and Central levels.

Establishing a dedicated training school with branches in all parts of the country for extensive training and retraining to the persons in the public services and users of NICNET.

Providing gateways for accessing international databases for those working in Districts, States and Centre.

Over the course of more than 25 years, NIC has established itself as a "primary builder" of e-government / e-Governance applications in the government sector (national, state, and local districts) as well as a champion of digital potential for sustainable development. With its ICT network, "NICNET," NIC has institutional connections with all of the Central Government's departments and ministries, 28 State governments, the National Capital Territory of Delhi, 6 Union Territories, and over 600 district administrations across India. NIC has played a key role in directing e-Government / e-Governance applications in government ministries/departments at the federal, state, local, and block levels. This has improved public services, increased transparency, encouraged decentralized planning and management, and increased accountability to the general public.

## **1.2 Technology Stack**

**Flutter:** flutter is google's portable ui toolkit for crafting beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

Flutter is built with c, c++, dart, and skia (a 2d rendering engine). See this architecture diagram for a better picture of the main components. For a more detailed description of the layered architecture of flutter, read the architectural overview.

Flutter is designed to support mobile apps that run on both android and ios, as well as interactive apps that you want to run on your web pages or on the desktop.

Apps that need to deliver highly branded designs are particularly well suited for flutter. However, you can also create pixel-perfect experiences that match the android and ios design languages with flutter.

Flutter's package ecosystem supports a wide variety of hardware (such as camera, gps, network, and storage) and services (such as payments, cloud storage, authentication, and ads).

**Dart:** Dart is a client-optimized language for developing fast apps on any platform. Its goal is to offer the most productive programming language for multi-platform development, paired with a flexible execution runtime platform for app frameworks.

Languages are defined by their *technical envelope*—the choices made during development that shape the capabilities and strengths of a language. Dart is designed for a technical envelope that is particularly suited to client development, prioritizing both development (sub-second stateful hot reload) and high-quality production experiences across a wide variety of compilation targets (web, mobile, and desktop).

Dart also forms the foundation of Flutter. Dart provides the language and runtimes that power Flutter apps, but Dart also supports many core developer tasks like formatting, analyzing, and testing code.

## **1.3 Introduction to Software**

The application have two modules such as Administrative and Citizen.

### 1)Administrative:

The Admin functionalities will include scheduling and Managing the Bookings of Raj Bhavan Places which includes Venue, Date, Time Slot, Fees, Allowed Visitors per Schedule, Approval of Visitors, Cancellation of Visit Schedule, etc by the Admin User of Raj Bhavan.

The Security Personnel Login will enable to check the visitors to the entry gate by scanning the QR Code and verifying the documents of the citizen.

### 2)Citizen:

The citizens will be able to book their visit to Raj Bhavan as per available seats on various allowed Time Slots, Upload Photos and Documents, and make an online payment. On Successfully booking citizens will be notified by Email, SMS, and In-app Notifications.

# **Visitor Management System for Raj Bhavan, Maharashtra (Mobile App Development)**

# **Software requirement specification**

# Visitor Management System for Raj Bhavan, Maharashtra (Mobile App Development)



**NATIONAL INFORMATICS CENTRE  
MAHARASHTRA STATE CENTRE, MUMBAI**

## **Restrictions:**

The application described in the document is furnished under a license agreement and may be used or copied only in accordance with the terms of the license agreement. It is against the law to copy the software on any other medium except as specifically allowed in the license agreement.

The licensee may make one copy of the software for backup purpose. No part of this software or document may be reproduced or transmitted in any form or by any means electronic or mechanical including photocopying and recording for any purpose without the written permission of NIC.

## **Warranty:**

This product is warranted for the period till the next changes are made in the application.

## **Liability:**

In no event NIC be liable for any damages, including loss of data arising out of the installation, maintenance, use, performance, failure or interruption of the software. Users are responsible to take backup of database on timely basis in suitable removable media.

<b>Prepared By</b>	Shalinikumari Prakashbhai Mane
<b>Reviewed By</b>	Shri AmodKumar Pradip Suryawanshi, Scientist-B System Analyst
<b>Approved By</b>	Mrs. Ireni Akojam, Scientist-F Senior Technical Director

## Revision

Version No.	Release Date	Change Number	Change Log
1	10 April 2022	Initial (1.0)	-

## History :

## Table of contents

<b>Section</b>	<b>Description</b>	<b>Page No.</b>
1	Introduction	19
1.1	Document Convention	19
1.2	Intended Audience and Reading Suggestions	19
1.3	Product scope	19
1.4	References	19
2	Overall Description	20
2.1	Product perspective	20
2.2	Product functions	20
2.3	User Classes and Characteristics	21
2.4	Operating Environment	21
2.5	Design and implementation constraints	21
2.6	User Documentation	21
2.7	Assumption and Dependencies	21
3	External Interface Requirement	22
3.1	User Interface	22
3.2	Hardware Interface	22
3.3	Software Interface	22
3.4	Communication Interface	22
4	System Features	23
4.1	OTP-based Login for Citizen	23
4.2	Dashboard	23
4.3	Schedule management	23
4.4	Visitor verification by QR	23
4.5	Booking by citizen	23
4.6	Report	24
5	Other Nonfunctional Requirement	24

## Visitor Management System for Raj Bhavan (Mobile App)

5.1	Performance Requirement	24
5.2	Safety Requirement	24
5.3	Security Requirement	24
5.4	Software Quality Attributes	25
5.5	Business Rules	25

## Introduction

### 1.1. Purpose

The purpose of the software requirement specification is to provide a clear, documented model of the requirements of a cross-platform Mobile Application of “Visitor Management for Raj Bhavan, Maharashtra”

The project is to develop the cross-platform (Android/iOS) mobile application of Visitor Management for Raj Bhavan, Maharashtra. The application will enable citizens to book their online Tickets by paying nominal fees to the Places of Raj Bhavan Maharashtra. The Application will empower the Administrator to manage and control the bookings. The verification of visitors at the entry gate based on a QR code will ease security personnel.

### 1.2. Document Conventions

This document follows the modern language association format. Bold Text has been used to emphasize the section heading. Highlighting the Text.

1. All text contained in this document is 11pt Arial font
2. Section titles are 18pt Arial font
3. Subsection titles are 14pt Arial font
4. Any further subsection breakdown is 12pt Arial font
5. All sections and subsections are numbered using the X.X.X... format, where X represents the number
6. Document text will be single-spaced and maintain 1" margins

### 1.3. Intended Audience and Reading Suggestions

This document is referred by the Project Manager, Project Developer, Project User, tester, etc.

### 1.4. Product Scope

- The software is for the administrator of Raj Bhavan Officials to manage the Visitors.
- Visitors can book a ticket as per seat and timeslot availability. On successful booking, the booking receipt with a QR code will be generated and sent to Visitor via e-mail, SMS, and In-App Notification.

- Security Personnel can verify the Visitor by scanning the QR code, in addition, security has access to the list of visitors for the next 3 (customizable by admin) upcoming bookings
- The Administrator, can create a schedule with custom date, time, No. of allowed visitors, fees, etc. The admin user can cancel the entire schedule, approve visitors, and access the various reports like date-wise list of visitors, datewise payments etc .

## 1.5. References

- 1) Existing Visitor Website of Raj Bhavan, Maharashtra  
<https://rbvisit.rajbhavan-maharashtra.gov.in>
- 2) GudApps Guidelines
- 3) Official Website of Flutter  
<https://flutter.dev/>
- 4) The IEEE format of SRS.

## 2. Overall Description

### 2.1. Product Perspective

This product aimed to enable the citizen to book tickets for Raj Bhavan visits and the Raj Bhavan Administrator to schedule/manage the visits and visitors through the mobile application. The security personnel can verify the visitors using QR Code scanning and access to visitor list through the mobile app.

### 2.2. Product Functions

- Visitor Management
- Time Slot management
- Visitor List
- Security management
- Documents Management
- Booking by Visitors

## 2.3. User Classes and Characteristics

**1) Visitor:** The visitor can book a ticket as per the available time slot to the various places of Raj Bhavan and get the booking receipt with a unique QR.

**2) Security Staff:** The security staff will verify the details of the visitors by scanning QR and allow visitors to get entry into Raj Bhavan.

**3) Admin:** Admin will create, manage or cancel the Visit Schedule by configuring the date, time, venue, fees, etc. Admin can approve Visitors, and send notifications to the visitors regarding any updates via email and SMS. The admin can access various reports like a datewise list of visitors, datewise payment details, and logs.

## 2.4. Operating Environment

The application is designed using flutter hence, it will be operated in Android 9 or above and ios 12 or above.

## 2.5. Design and Implementation Constraints

- 1) User must have OS with minimum Andriod 9 or iOS 12
- 2) User must have good network access to run the application
- 3) User required to grant the access of Camera, File System, Internet for the functioning of an application.
- 4) The APIs must be always accessible for data exchange over a secure environment

## 2.6. User Documentation

The help/user documentation for citizens will be available in an application under the help section, which will have stepwise screen-by-screen guidelines for visitors like documents required to upload, and keep card details / UPI details handy for payment.

The user documentation for Admin / Security Personnel will be available under their respective logins or shared with authority by email which will have the details about how to use the application.

## 2.7. Assumptions and Dependencies

- The visitor, admin, and security have the knowledge of how to use the mobile application.
- Good Internet connectivity is required to use the application.

- APIs are always accessible
- Device permissions (Camera, File System, Network) are granted for the application

### 3. External Interface Requirements

#### 3.1. User Interfaces

The main screen is the dashboard for security. That is by default home page of the security.

1)Dashboard: The User interface is good and easy to use by the security. In that, The earlier visiting list is displayed to the security. Security can see the upcoming three days visiting lists.

2)QR Scan: security can scan the QR scan to get the visitor details to allow the entry of visitors into the RAJ BHAVAN.

3)Visitor List: The visitor list is displayed on this page. Security can access the whole list of visitors with the date, time, and documents. security personnel has to select the venue name, date, and time to check the details of the visitor.

#### 3.2. Hardware Interfaces

##### Hardware:

- Processor: Minimum Core i3 Processor
- RAM: Minimum 8GB
- 256 GB SSD
- Android Mobile Phone

#### 3.3. Software Interfaces

##### Technology Stack:

- Microsoft Visual Studio Code 1.66
- Flutter 2.10.3
- Android Studio 2021.1.1

#### 3.4. Communications Interfaces

- Application required Good Network for functioning
- Application will access the APIs over SSL

## 4. System Features

### 4.1. OTP-based login for Citizen

- OTP-based login with email or mobile number for Citizen, hence no need to remember credentials.
- Username and Secure Password-based login with 2FA for Rajbhavan Administrator and Security

### 4.2. Dashboard

1. **Citizen** dashboard shows the list of upcoming bookings, books from the account
2. **Admin** and Security dashboard shows the line charts (Datewise) to monitor the number of visitors

### 4.3. Schedule Management

The following schedule management can be performed by the Admin of Raj Bhavan.

#### 1.Create a schedule by configuring :

- Venue: From Multiple available Venue
- Date: Multiple Date can be selected
- Time: From Time and To Time
- Number of allowed visitors: maximum allowed visitors
- Fees: Variable fees for each Schedule
- Approval: optional for auto-approval

#### 2.Cancel the schedule

#### 3.Approve Visitors

#### 4.Send Notification to the Visitors

### 4.4. Visitor Verification by QR

1. Each visitor will have a unique QR on Booking Receipt
2. Security Personnel / Admin of the Raj Bhavan can verify the visitor by scanning the QR code.
3. On successful scanning of the QR code, visitor details will be displayed with photos and uploaded documents which will make the process smooth and faster than before

### 4.5. Booking by Citizen

Citizen can book visits to the Raj Bhavan Places in 4 steps

Reserve Number of seats (max 10), Personal Details, Upload Documents and Payment

## 4.6. Reports

Admin can view the various reports like Datewise List of Visitors, Datewise Payment Received, Logs etc.

### 4.1.2 Stimulus/Response Sequences

Admin:

### 4.1.3 Functional Requirements

Login: Admin and Security can log in to check the visitor details and approval.

Visitor List: Security can check day-wise visitor list

Visiting time slot: Admin can allow the visiting hours.

QR Scan: security can check the visitors details with their proper documents.

## 5. Other Nonfunctional Requirements

### 5.1. Performance Requirements

The throughput and latency of the application are dependent on Network Speed and mobile device. Good Network connectivity and mobile device is recommended and high-performance output.

### 5.2. Safety Requirements

- Proper version management is required before release of application
- Application must be tested on various devices to ensure it should not crash
- Application must be responsive even if the network goes disconnected while using the application
- Source Code and APK must be backed up in safe vault

### 5.3. Security Requirements

The application is already taken care of all known top vulnerabilities, however, updates may be required in the future to address the new vulnerabilities. The Web API

must be hosted in a secure environment over the SSL

system will use a secure database. user can only use the application. There is authentication for security and admin roles.

## 5.4. Software Quality Attributes

1. **Correctness:** The application ensures the correctness of data by validating each input field strictly
2. **Authenticity:** The Visitors are verified by OTP authentication.
3. **Adaptability:** The application is designed by considering the best UI/UX experience for the user, hence it can be adoptable quickly.
4. **Availability:** The application will be available in major app stores/platforms for users to download and use.
5. **Maintainability:** The application is developed module-wise and can be customized at any time by just configuring a few parameters

## 5.5. Business Rules

There are 2 types of users who can use this application admin and security. The admin have all the permission to control the application.

## Other Requirements

All requirements are covered in the above topics. There are no other requirements.

## Appendix A: Glossary

Admin

Security

Payment

Visitors

## **Appendix B: To Be Determined List**

- 1) Visitors can be tracked during their stay inside Raj Bhavan by accessing the visitor's live location.
- 2) Any urgent alert or message can be notified to the visitor if the visitor is entering a restricted area.
- 3) Visitors and accompanied can be verified using face recognition at the security gate along with QR Code.
- 4) Predictions of the Number of Visitors based on historical and timeframe data such as expected visitors during the weekend, festivals, etc.

# **Software Design Document**

# Software Design Document

<b>Section</b>	<b>Description</b>	<b>Page No.</b>
	Revision History	31
1	Introduction	31
1.1	Document Purpose	31
1.2	Product Scope	31
1.3	Definitions, Acronyms and Abbreviations	32
1.4	References	32
1.5	Document Overview	32
2	Product Overview	33
2.1	Product perspective	33
2.2	Product functions	33
2.3	Product Constraints	34
2.4	User Characteristics	34
2.5	Assumptions and Dependencies	34
3	Requirements	35
3.1	External Interfaces	35
3.1.1	User Interface	35
3.1.2	Hardware Interface	39
3.1.3	Software Interface	40
3.2	Functional	40
3.3	Quality of Service	40
3.3.1	Performance	40
3.3.2	Security	40
3.3.3	Reliability	40
3.3.4	Availability	41
3.4	Compliance and Implementation	41

## Visitor Management System for Raj Bhavan (Mobile App)

3.5	Design	41
3.5.1	Installation	41
3.5.2	Distribution	41
3.5.3	Maintainability	41
3.5.4	Reusability	41
3.5.5	Portability	42
3.5.6	Cost	42
3.5.7	Deadline	42
3.5.8	Proof of concept	43
4	Verification	43
5	Appendices	44

## Revision History

Name	Date	Reason For Changes	Version
Shalinikumari Mane	10-April-2022	Initial	1.0

# 1. Introduction

Software Design Document of Raj Bhavan Maharashtra Mobile Application will help the developer and stakeholders like System Analyst, Designer, Developer, Tester etc. to plan the workflow and workload and agree upon their actions in order not to do unnecessary work.

This document covers design aspects of the system in respect of hardware resources required for development and deployment, software interfaces, and User Interfaces throughout the lifecycle of Product.

The project is to be develop a cross-platform (Android/iOS) mobile application of Visitor Management for Raj Bhavan, Maharashtra. The application will empower citizens to book visit at Raj Bhavan through mobile application by paying nominal fees. The Application will empower the Administrator to manage and control the bookings and visitors. The verification of visitors at the entry gate based on a QR code will ease security personnel.

## 1.1 Document Purpose

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to build. This document ensures design specs of the software are understood and it's clear to all and given clear ideas to the development team to build the system rapidly. It specifies what is possible with the product and how it can be accomplished. The intended audience for this document is project manager, system analyst, designer, developer, tester.

## 1.2 Product Scope

The project scope will be to develop the cross-platform Mobile Application to manage the Bookings of Raj Bhavan Places including the development of APIs.

There are 3 different roles of the users of application viz. Citizen, Administrator and Security Personnel.

Administrator functionalities will include Schedule, Cancellation and Managing the Bookings of Raj Bhavan Places. The customized parameters for scheduling of bookings will be venue (Master Data), date, time slot i.e. start time and end time, fees (including free schedules), number of visitors allowed, approval of visitors.

Security Personnel wil have limited functionalities, just to check the visitors at the entrance by scanning the QR Code and verifying the documents upto 3 schedules (number will be customizable by admin).

Citizens can book their visit to Raj Bhavan Places as per available seats and time slots, upload photos, and documents, and make an online payment. On successful booking, citizens will be notified by Email, SMS, and In-app Notifications.

## 1.3 Definitions, Acronyms, and Abbreviations:

### Abbreviations:

SDD	Software Design Document
VM	Virtual Machine
SMS	Short Message Service
QR	Quick Response Code
API	Application Programming Interface
JSON	Javascript Object Notation
SSL	Secure Sockets Layer
UAT	User Acceptance Testing
SSD	Solid State Drive
PoC	Proof of Concept
GB	Gigabyte

## 1.4 References

- 1) <https://maharashtra.nic.in>
- 2) <https://rajbhavan-maharashtra.gov.in>

## 1.5 Document Overview

Rest of the document organized as Section 2 covers Project overview comprises with product perspective, functions, constraints etc. section 3 covers Requirement details comprises with external interfaces, Hardware and Software interfaces, functional aspects, quality of service, design and implementation etc. section 4 covers verification methodologies.

## 2. Product Overview

- o To schedule and manage the Bookings of Raj Bhavan Places which includes venue, date, time slot, fees, and number of allowed visitors.
- o To cancel the schedule
- o To get the Reports on visitors, payments, SMS Logs, etc.
- o To verify the visitor at entry gate by scanning the QR Code.
- o To get the list of Visitors for the next 3 (customizable by admin) schedules.
- o To enable citizens to book their Raj Bhavan Visits for various places as per available dates and time slots.
- o To send notifications to the citizens via Email, SMS, and in-app notifications.

The project scope will be to develop the cross-platform Mobile Application to manage the bookings of Raj Bhavan Places including the development of APIs.

### 2.1 Product Perspective

This product aimed to enable the citizen and Raj Bhavan Administrator to manage and book a ticket for RAJ BHAVAN through the mobile application from any time anywhere. The visitor can book the ticket as per the available time slot and seat availability through the mobile application. The security personnel can also check the visitor list through the mobile app.

### 2.2 Product Functions

- Visitor Management
- Time Slot management
- Visitor List
- Security management
- Documents Management

## 2.3 Product Constraints

The mobile application is dependent on the APIs for smooth functioning. The API must provide the proper success, failure and error message to process the data on mobile application.

The user must have android 9 and onwards and iOS 12 onwards.

The user needs to approve following permissions to the app

- Network : To access the internet
- Camera : To scan the QR
- File Storage : To store authentication token in secure storage.

The payment gateway must respond within timeout for successful booking

## 2.4 User Characteristics

**1) Visitor:** The visitor can book a ticket for visiting RAJ BHAVAN and get the unique QR for visit.

**2) Security Staff:** The security staff will verify the details of the visitors using scanning QR and allow visitor to get entry in Raj Bhavan.

**3) Admin:** Admin will create and manage the Visit Schedule and Visitors. Admin can approve visitors. The admin can configure time slots, fees, number of visitors etc.

## 2.5 Assumptions and Dependencies

Assumptions :

- 1) Web APIs will provide status of request ie. success, failure and error
- 2) If status is failure or error, the proper error / failure message will be returned by API
- 3) If status is success the data in predefined json format will be returned
- 4) Web APIs will be hosted under SSL for secure communication
- 5) All the required permissions like camera, storage will be granted by user

Dependencies :

- 1) Application uses many third party libraries for rapid development such as QR Scanner.
- 2) Network must be stable with consistent high quality bandwidth
- 3) User of the application must grant "allow to install app from unknown source" for testing and UAT purpose

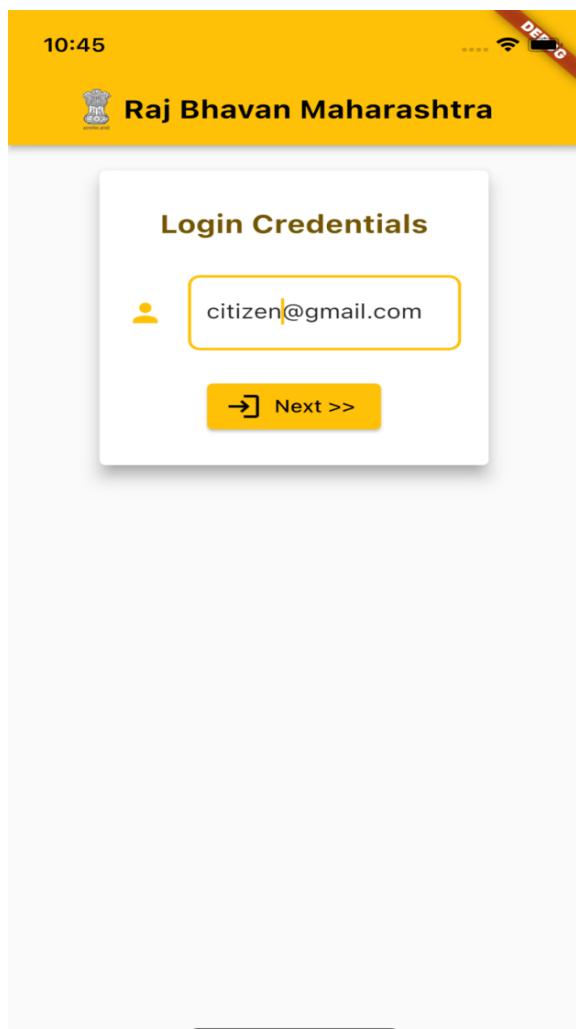
## 3. Requirements

### 3.1 External Interfaces

#### 3.1.1 User interfaces

Login screen : On login screen, it will accept username, mobile number and email Id in username field depends on the pattern mobile number or email Id will be considered as citizen login and username in alphabets will consider as Official Login (either admin or security personnel)

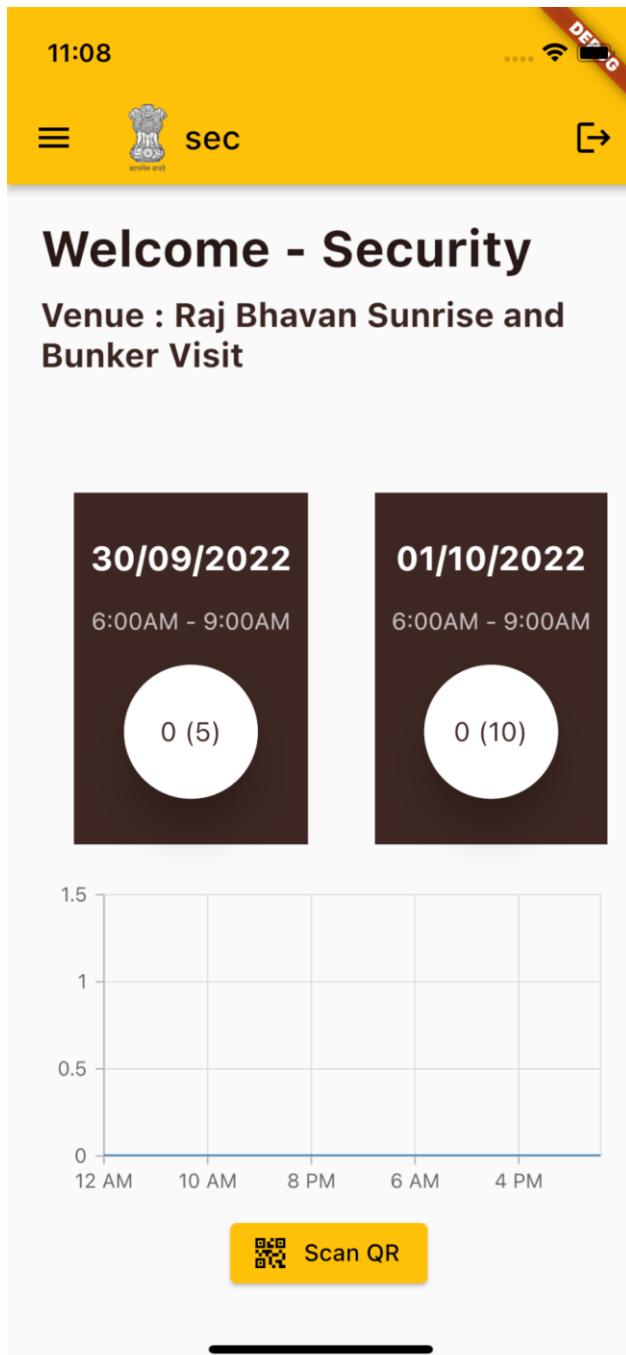
Depending on the username pattern it will give next screen to enter for OTP for citizen or password for department officials



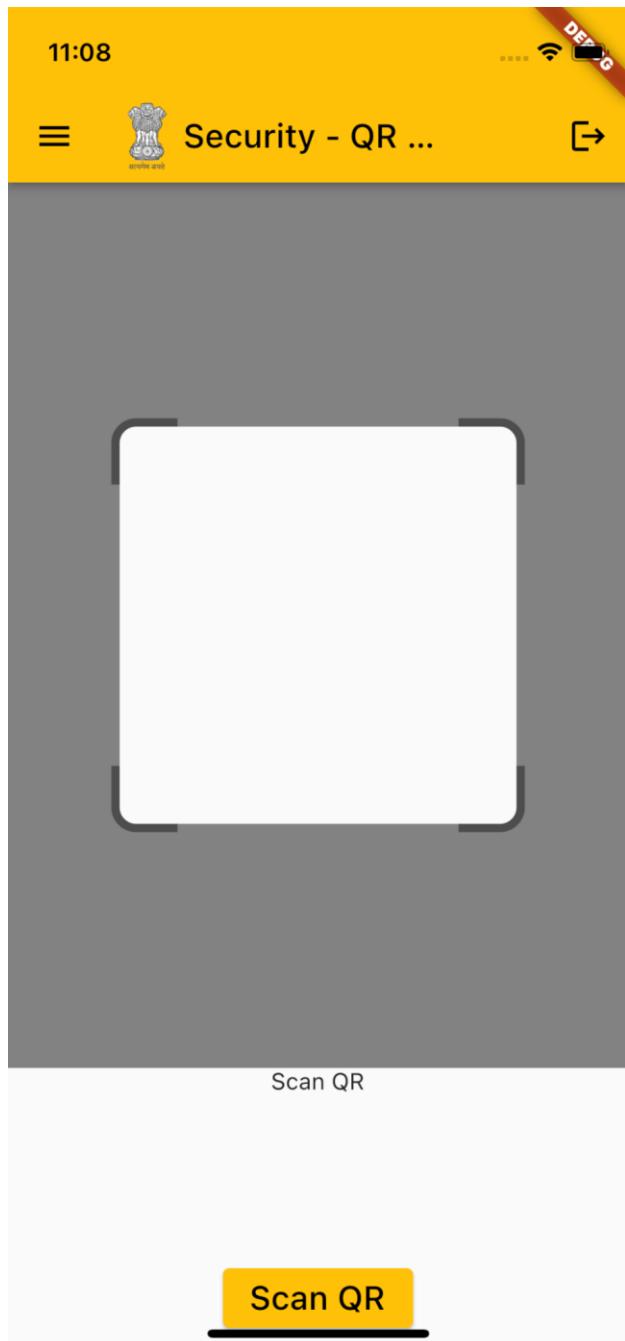
After successful login dashboard screen will appear on screen (customized for each role)

- 1) Dashboard:
  - a) Security can see three (customizable by admin) upcoming visit schedule with line charts.

- b) Admin can see all upcoming visit schedules with line charts.
- c) Visitor can see his upcoming booking or button to book the visit

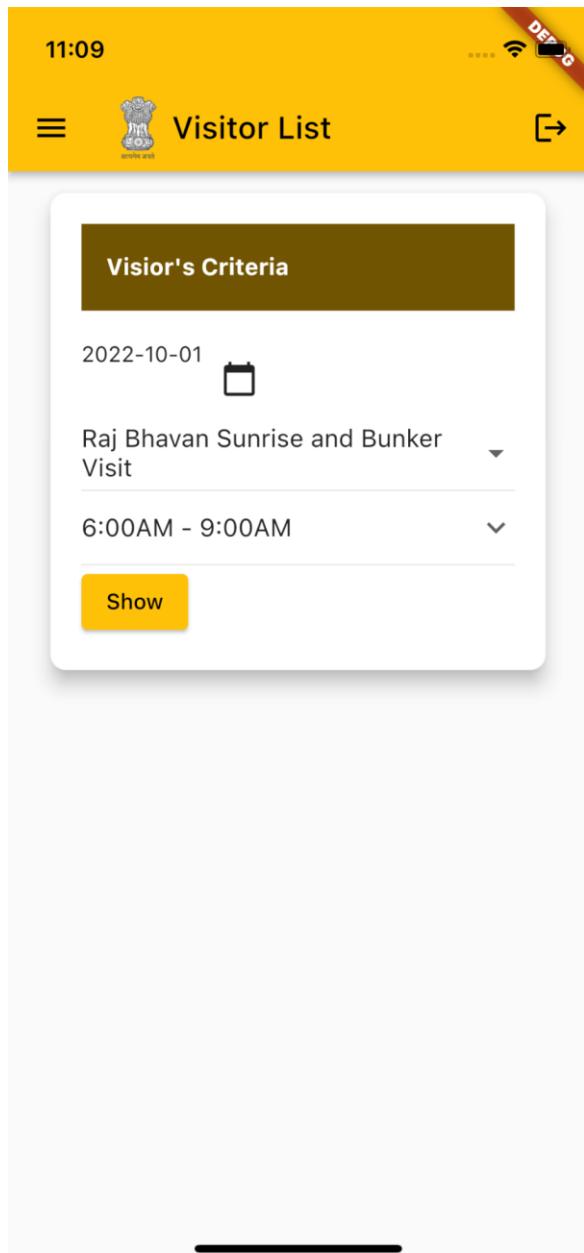


- 2) QR Scan: This option is enabled for security and administrator which is useful to scan the QR scan to get the visitor details to allow the entry of visitors into the RAJ BHAVAN.



- 3) Visitor List: This option is enabled for administrator and security (for predefined count of upcoming schedule, default 3) to get the list of visitors with date, time, and documents.

## Visitor Management System for Raj Bhavan (Mobile App)



- 4) Payment Details : This option is enabled for administrator to get the datewise payment report.

Date	Amount
04-05-2022	3750.0
05-05-2022	1950.0
06-05-2022	4500.0
07-05-2022	5700.0
08-05-2022	4500.0
10-05-2022	2700.0
11-05-2022	300.0
12-05-2022	2250.0

The booking to the Raj Bhavan places is available through citizen login comprising of 4 steps i.e. Check and reserve the number of seats (upto 10) as per availability of respective venue and time, Personal Details (name, age, gender, email id, mobile number, vehicle details) and companion details (name, age, gender), Photo and Identity document uploading and Payment.

### 3.1.2 Hardware interfaces

#### Development:

##### Hardware:

- Processor: Minimum Core i3 Processor
- RAM: Minimum 8GB
- 256 GB SSD

#### End-User Requirements:

- Android / iOS Mobile.

### **3.1.3 Software interfaces**

#### **Technology Stack:**

For Mobile App development:

- Microsoft Visual Studio Code 1.66
- Flutter 2.10.3
- Android Studio 2021.1.1

## **3.2 Functional**

1 : The latest mobile device is recommended as applications needs to access Camera to scan the QR Code, GPS for geo fencing.

2 : Email and SMS services must be enabled on Web VM to send the notifications by Email and SMS.

3 : Mobile device must have the Network connection.

4: The necessary permission must be granted to the application to access the device specific functions such as Camera, GPS, Internal Storage

## **3.3 Quality of Service**

### **3.3.1 Performance**

The throughput and latency of the application is dependent on Network Speed and mobile device. Good Network connectivity and mobile device is recommended and high performance output.

### **3.3.2 Security**

The application is already taken care of all known top vulnerabilities, however, updation may be required in the future to address the new vulnerabilities. The Web API must be hosted in secure environment over the SSL.

### **3.3.3 Reliability**

All the standard guidelines are to be followed while developing the application which prove the reliability on application

### 3.3.4 Availability

The sufficient efforts to be taken to securely store the data in devices, transit and at rest (Database). In case of change of device or factory reset, the data from server will be fetch and made available at device.

## 3.4 Compliance

- Report format : Reports are generated and displayed on screen and can be exported into PDF.
- Data naming : standard naming conventions to be used for json data exchange. The variables to follow camelCase.
- Accounting procedures : Payment information is stored securely for accounting purpose.
- Audit tracing : User Registration, Login, SMS, Payment details are recorded into audit trail for tracing and tracking.

## 3.5 Design and Implementation

### 3.5.1 Installation

The Application can be downloaded from App Store for iOS and google play store for android devices. The minimum version for Andriod is 9 and iOS is 11

### 3.5.2 Distribution

The Application can be hosted on google play store for android devices and app store for iOS devices, to make it available on public platforms. However signed bundled apk may be distributed among Office (Admin and Security) use.

### 3.5.3 Maintainability

The Application must be maintained and updated regularly if found any new vulnerabilities in dependent packages. However, package compatibility must be checked on sandbox before release on production environment.

### 3.5.4 Reusability

The application will be developed with maximum configuratble parameters as below:

- **Office** : Any Office can use the application by just configuring Office details.
- **Venue** : Office can add multiple venues across any geographical locations

- **Date and Time slots** : Venuewise date and time slot (from time and to time) can be configured.
- **Number of Visitors** : Number of allowed visitor per timeslot can be configured
- **Fees** : Customized fees can be configured with optional details as senior citizen, children, handicap etc. or fees can be set as 0 or free visits

The application to be developed considering with the generic requirements and product base approach, hence it can be reuse for any Office and Department.

### **3.5.5 Portability**

The flutter is to be used for development as it is natively compiled, multi-platform support i.e. Andriod, iOS, Desktop App and Web App with a single codebase

### **3.5.6 Cost**

The software is being developed using opensource i. e. flutter so there will be no software and development cost involved. However, hosting the app on google play store for android and app store for iOS may incur the cost as per policy of respective providers.

### **3.5.7 Deadline**

T: Timeline starts from day, figure below are shown in weeks i.e T+1 means deadline for 1 week from the start date

T : Start of project

T+1 : Software Requirment Specificatin

T+2 : Software Design Document

T+3 : Setup of Development Entionment

T+4 : Login and Authentication

T+5 : Role based Dashboards

T+10 : Citizen Booking Module

T+14 : Admin Create Schedule Module

T+16 : Security Personel Module

T+18 : Reports

T+20 : Testing and UAT

### **3.5.8 Proof of Concept**

To test the proof of concept old or dummy data may be required. Different devices with different configuration will be ideal for PoC.

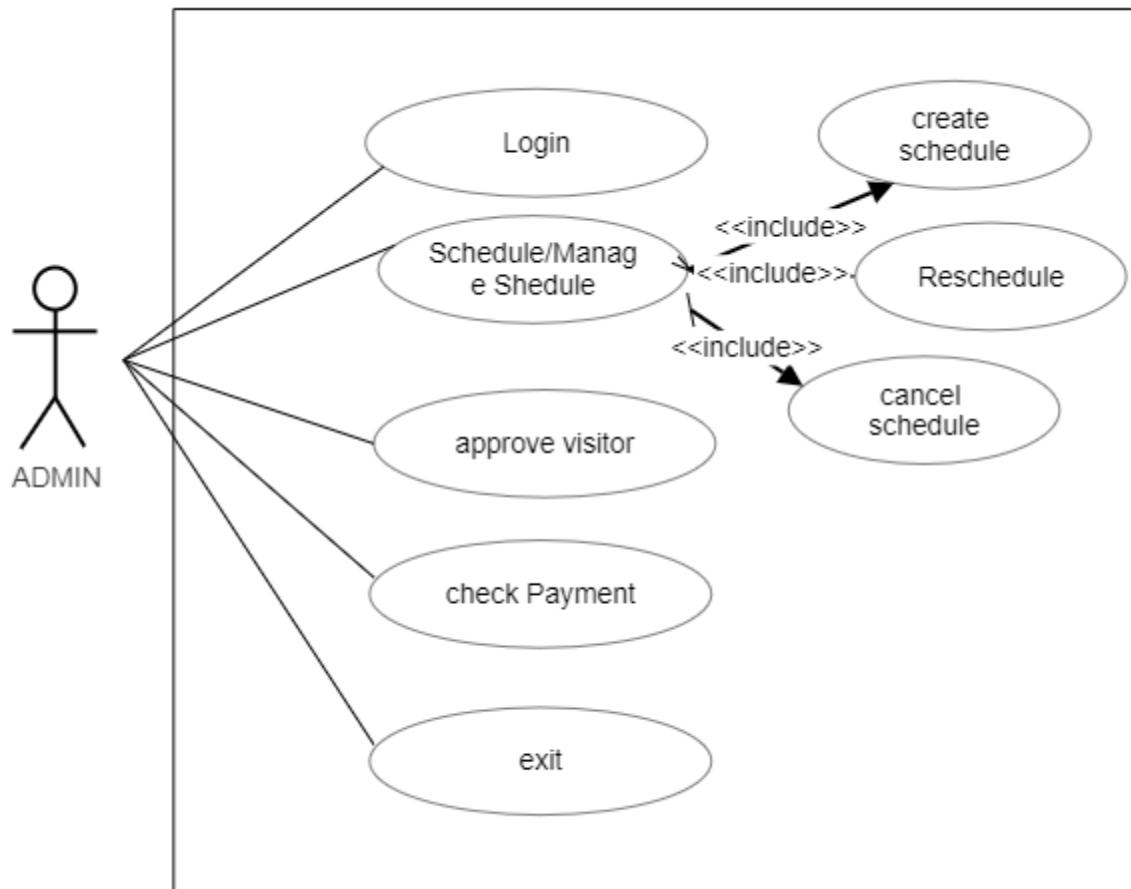
## **4. Verification**

The throughput and latency of the application is dependent on network speed and mobile device. Good Network connectivity and mobile device is recommended and high performance output.

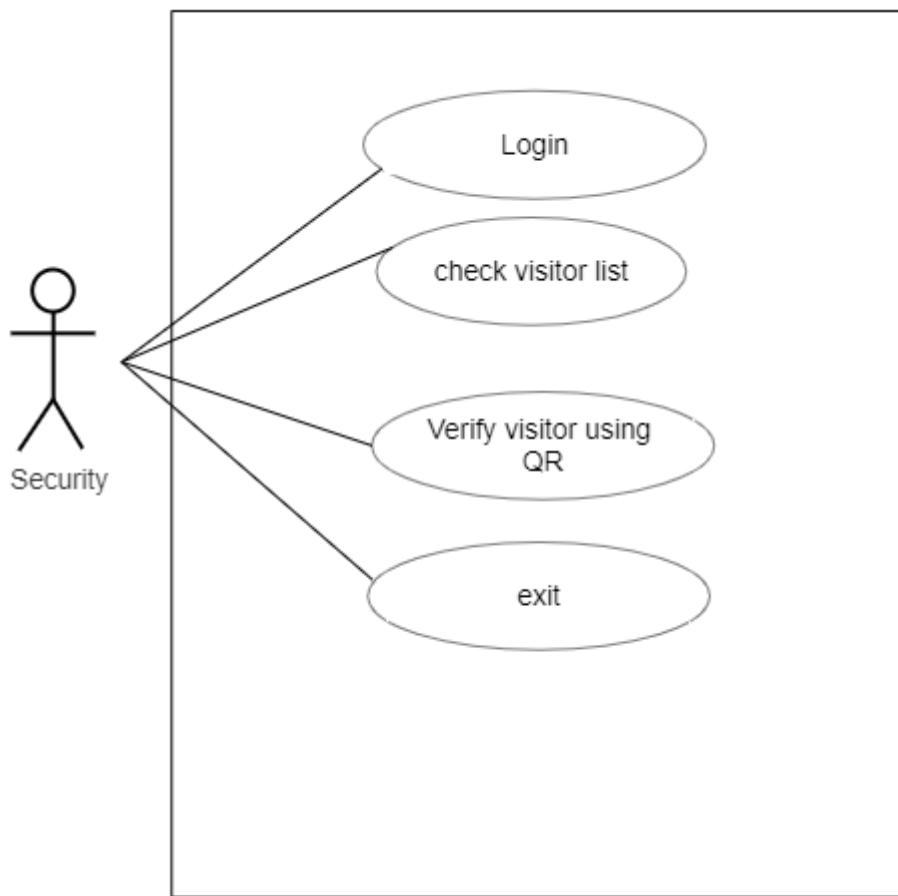
## 5. Appendixes

### USE CASE DIAGRAMS

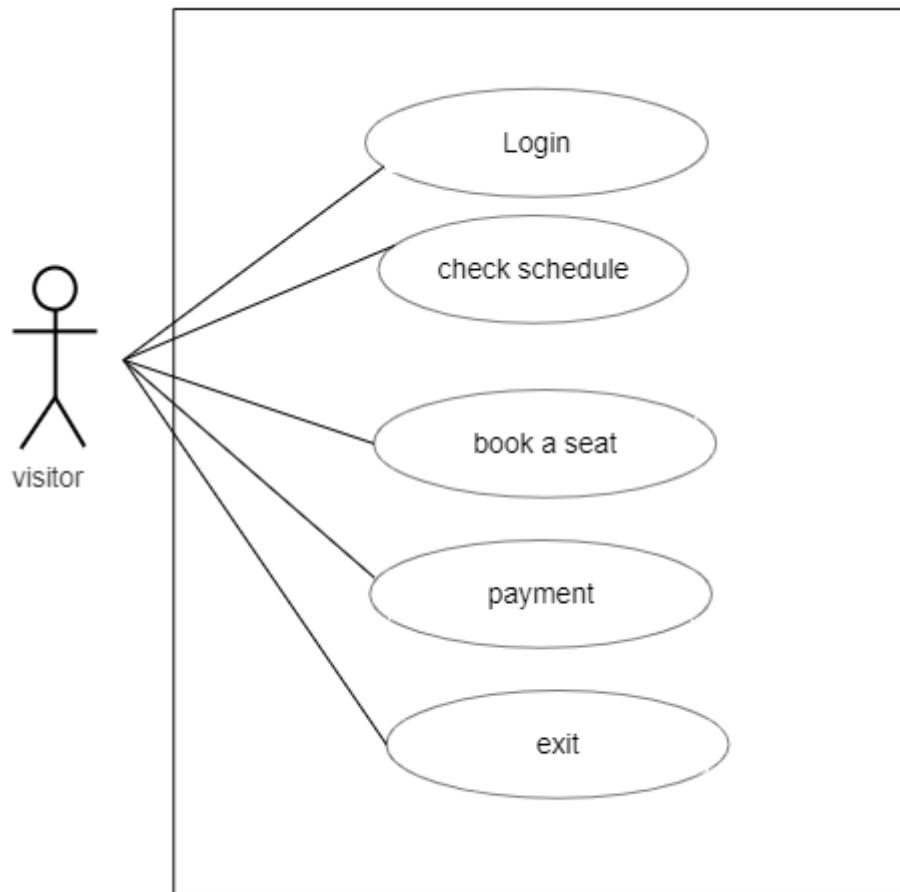
**Admin:**



## Security:



## Visitor:



# User Manual

**Table of Contents:**

<b>Section</b>	<b>Description</b>	<b>Page No.</b>
1	Introduction	49
2	Hardware and software specification	49
3	Installation/Setup Guide	49
4	Main Features	49
5	Raj Bhavan Officials Module	50
6	Citizen Module	58

**Introduction:**

This document is useful for the users of the mobile application of Raj Bhavan. The intended users for this document are Raj Bhavan Officials and Citizens.

This document is broadly divided into two sections

- 1 )User Manual for Raj Bhavan Officials
- 2) User Manual for Citizen

**Minimum Hardware And Software Requirement Hardware:**

The Mobile application required Andriod version 8.0 (Oreo) and above with camera, internet, and location access permission.

**Installation/Setup Guide:**

The application can be installed with apk (from Play Store in the future).

**Main Features:**

- 1) Role based Login
- 2) Ease of use for administrator to manage the visitors and visit schedule
- 3) Ease of use for Citizen to book their visits for Raj Bhavan
- 4) QR based entry eases verification of visitors
- 5) Customizable Reports available for Raj Bhavan Officials
- 6) Graphical representation of datewise visitors useful for trend analysis.

## User Manual for Raj Bhavan Officials:

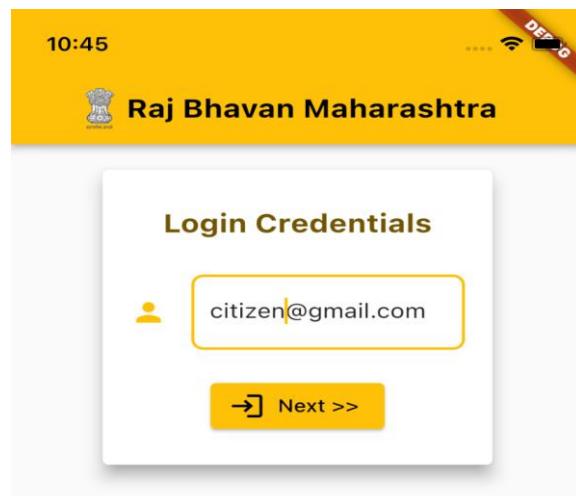
After successfully installation of APK, Raj Bhavan Officials can log in through their existing web application credentials.

### ADMIN:

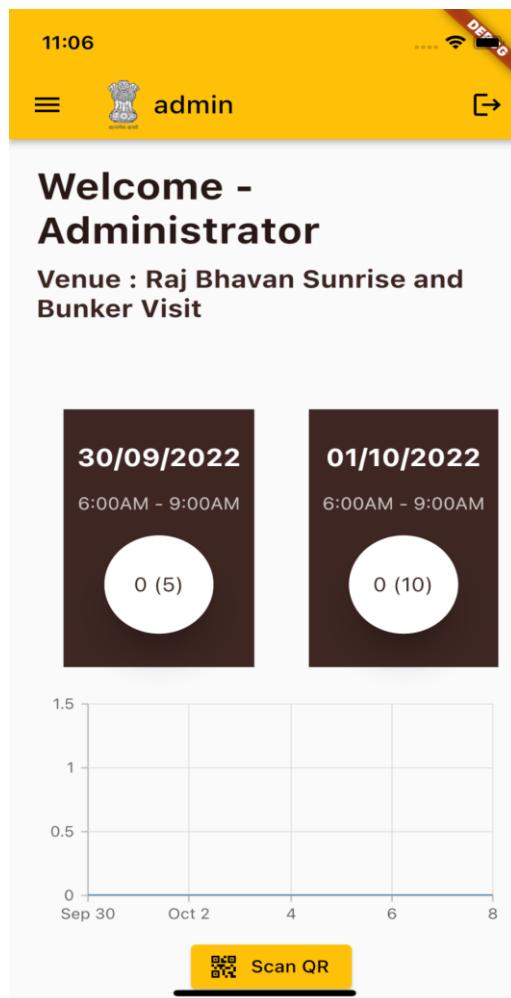
#### 1) LOGIN:

Raj Bhavan Officials (Admin Role and Security Personnel Role) can log In to Application using the existing credentials

- 1) The App will open with a Login screen (if not logged in previously else it will open the dashboard screen).
- 2) Enter username
- 3) The next screen will appear will be for password
- 4) Enter the password and click on the Login button
- 5) If user is authenticated successfully, the dashboard screen will be shown on screen else, it will show an appropriate error message



2) Dashboard: On successful login of the user (admin or security personnel role) It will show the upcoming Booking as per permissions on dashboard screen with the graphs.



The top side menu bar comprises with options as below

<b>Admin Role</b>	<b>Security Personnel Role</b>
Home	Home
QR scan	QR Scan
Create Schedule	Visitor List
Visitor List	Logout
Payment Report	
Logout	

**Home**:- The Home options will open the dashboard screen as described above

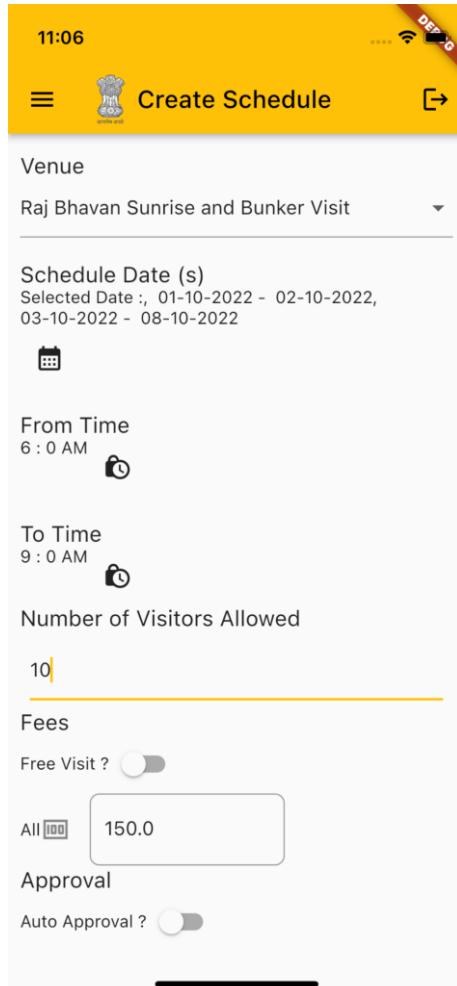
**QR Scan**: This option is available for both the roles to scan the QR Code of the visitor to allow for the entry and verification of visitor.

The first time when we open this option, it may ask to grant permission to access the camera. So the user may choose While Running App Permission for the camera.

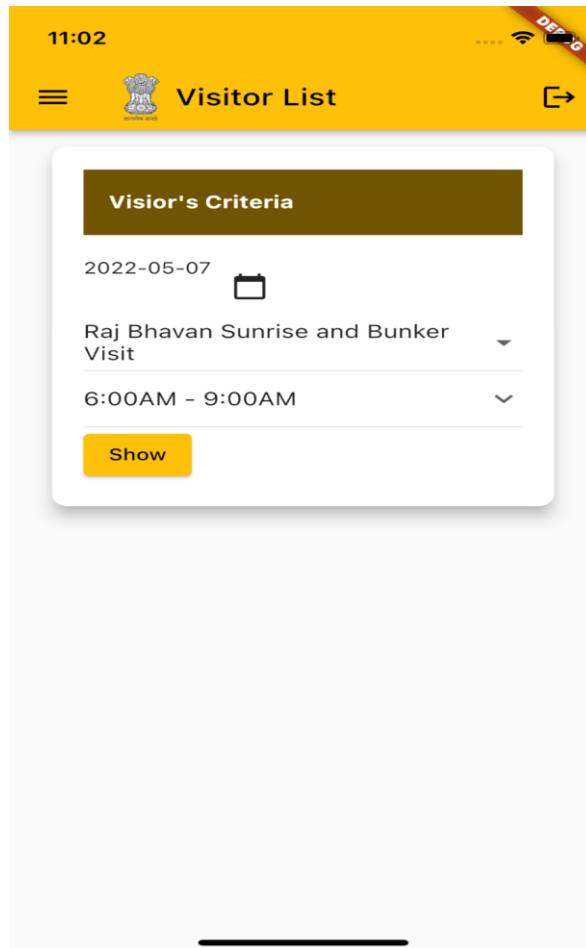
After granting the first-time permission, it will open the camera and look for the appropriate QR. Once the desired QR is scanned it will display the visitor details on the screen.

**Create Schedule:** The admin can create a schedule by configuring the mandatory fields such as Venue(Sunrise gallery/Bunker Visit), Schedule Date (Multiple date can be selected), From Time To Time, Number of visitors to be allowed, Fee (or Free visit), Approval (Auto or manual).

It will open the booking slot for visitors as per the schedule configured by the admin.



**Visitor List:** The admin can see the visitor list by selecting the Date, Venue, and Timing. Click on the show button will open the next screen with the visitor list.



**Payment Report:** The admin can see the Payment details of each visitor by giving the period (from date and to date), Select the venue and click on the Show Button.

payment details of specified period with the total amount will be display on screen.

The screenshot displays two identical screens of a mobile application titled "Datewise Payments".

**Left Screen (Criteria):**

- Header: "Datewise Payments"
- Section: "Payment's Criteria"
- Date Range: "From 2022-05-01" and "To Date 2022-05-31" (with calendar icons)
- Venue: "Raj Bhavan Sunrise and Bunker Visit" (with a dropdown arrow)
- Button: "Show" (highlighted in yellow)

**Right Screen (Results):**

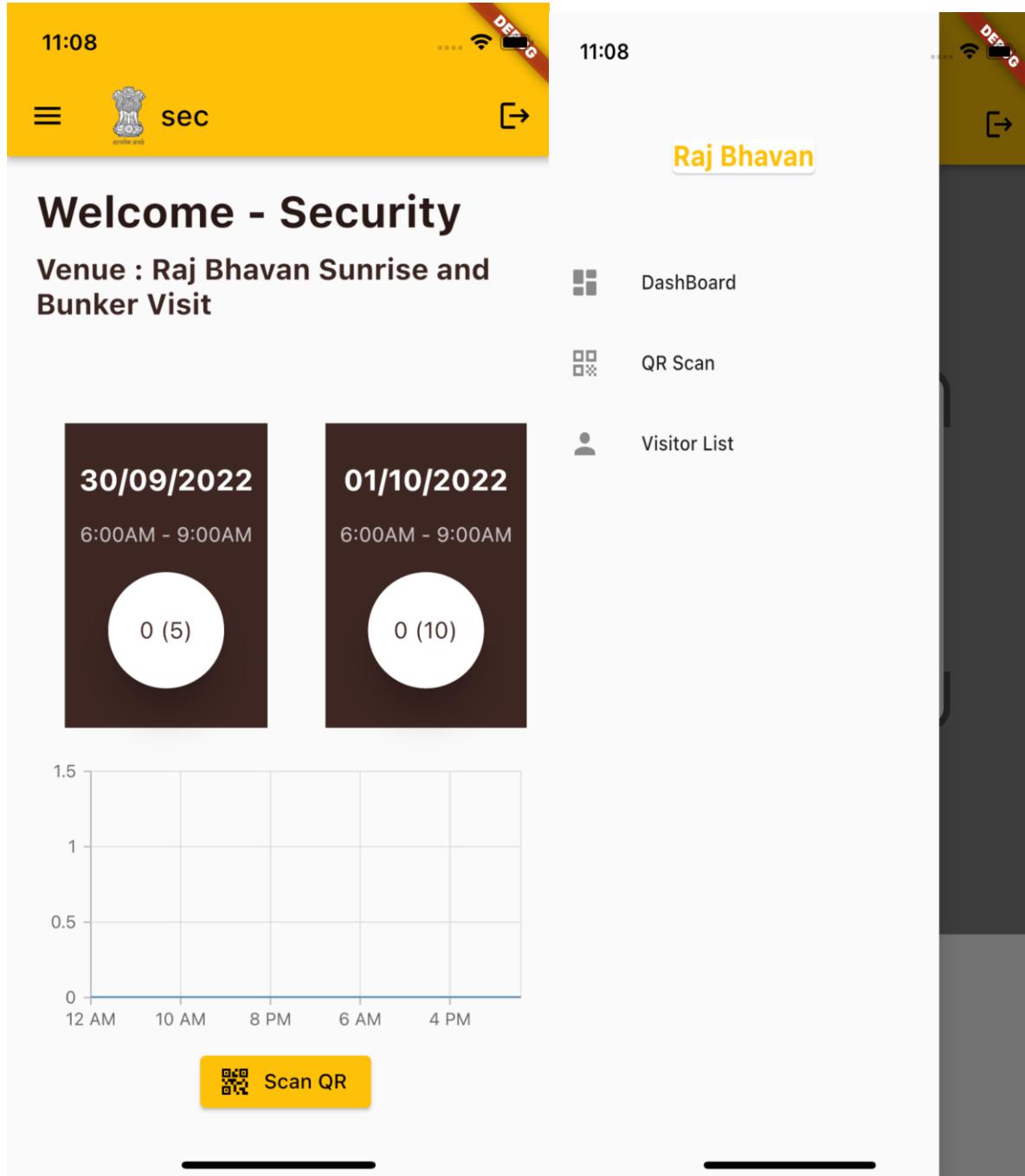
- Header: "Payments" and "From / To Date : 2022-05-01 2022-05-31"
- List of payments:

  - 04-05-2022 Amount : 3750.0
  - 05-05-2022 Amount : 1950.0
  - 06-05-2022 Amount : 4500.0
  - 07-05-2022 Amount : 5700.0
  - 08-05-2022 Amount : 4500.0
  - 10-05-2022 Amount : 2700.0
  - 11-05-2022 Amount : 300.0
  - 12-05-2022 Amount : 2250.0

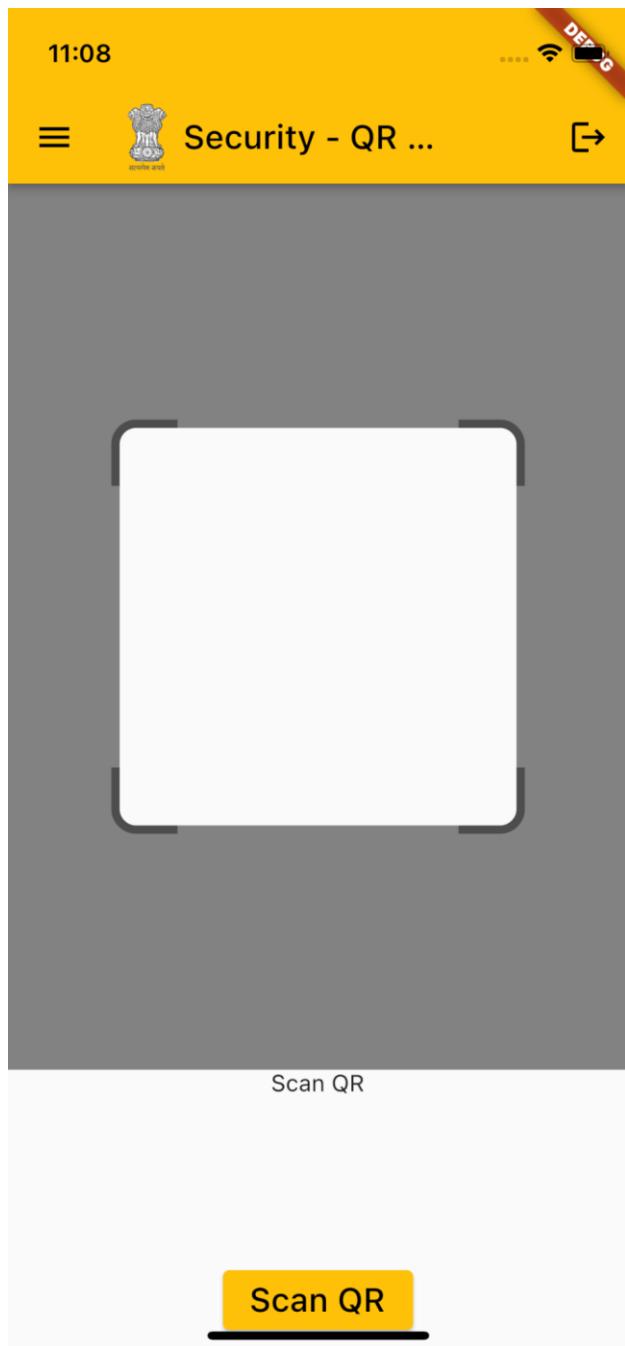
Security:

The security personnel will have the following limited features:

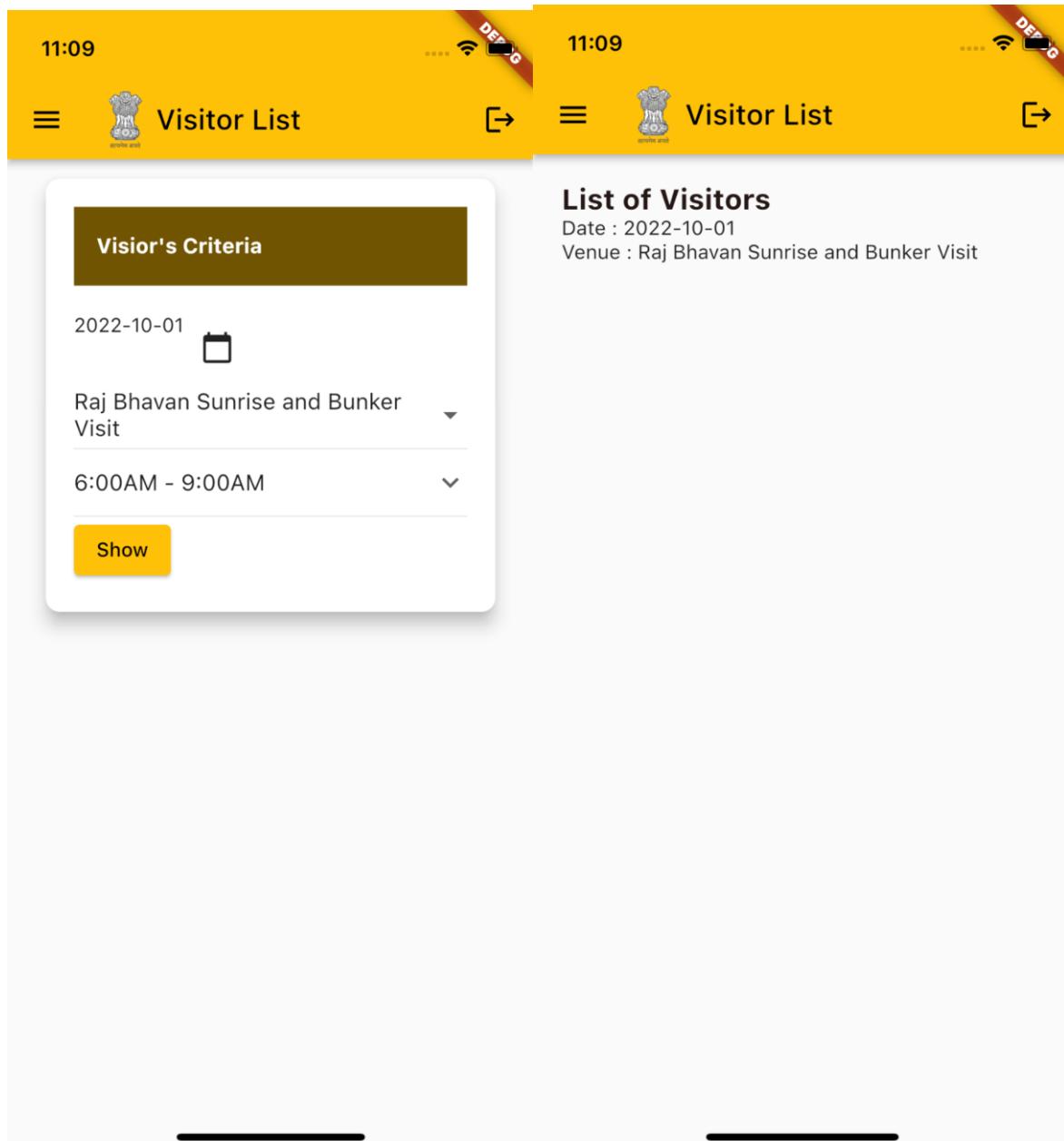
**Dashboard:** Security Personal can see the next upcoming visitor's List and can see the Menus.



**QR Scan:** Security Personal can check the visitor's detail by scanning the QR code of the visitor.



**Visitor List:** Security personnel can see the visitor list and their details for security purposes.

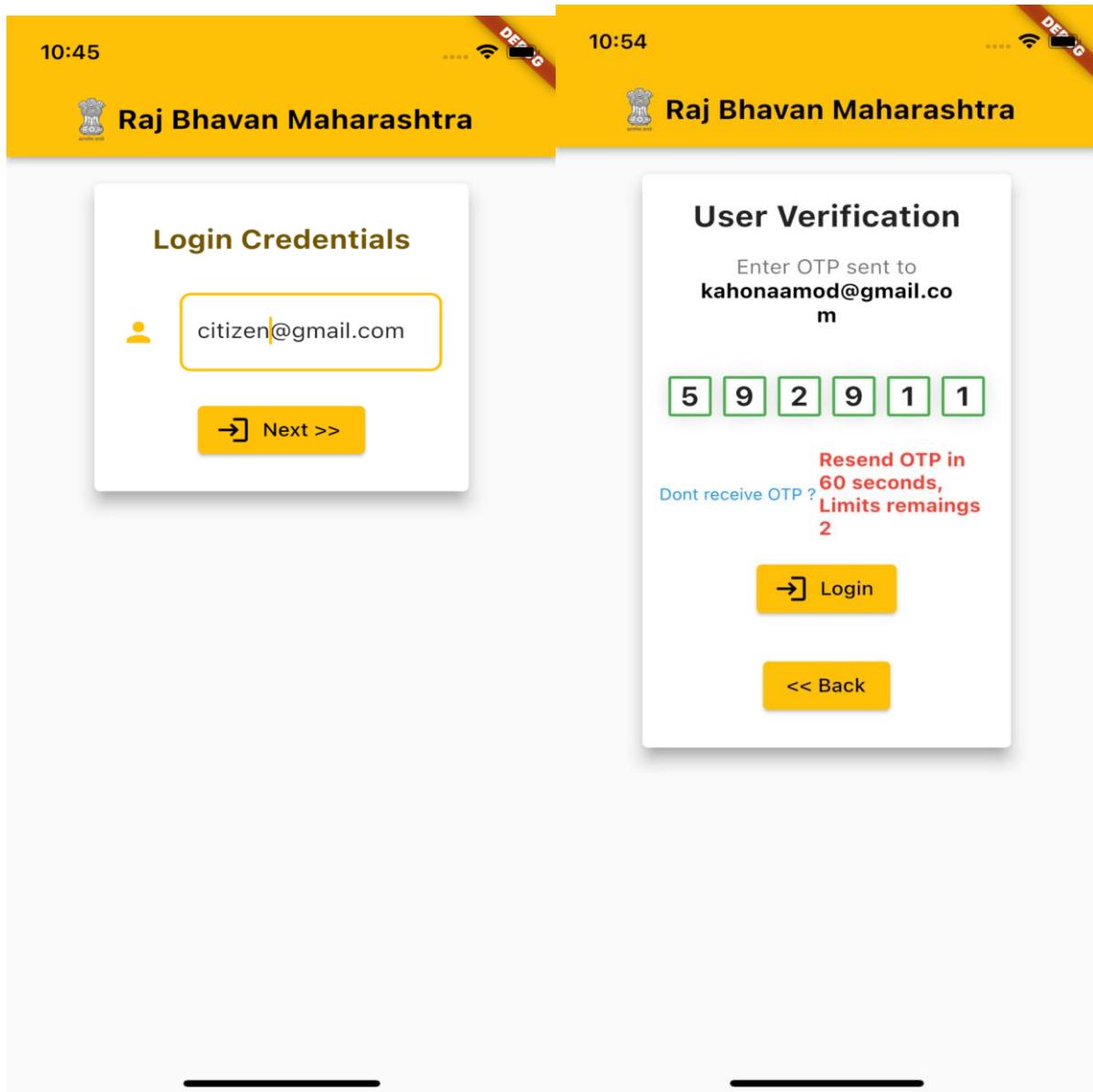


## User Manual for Citizen:

### Citizen:

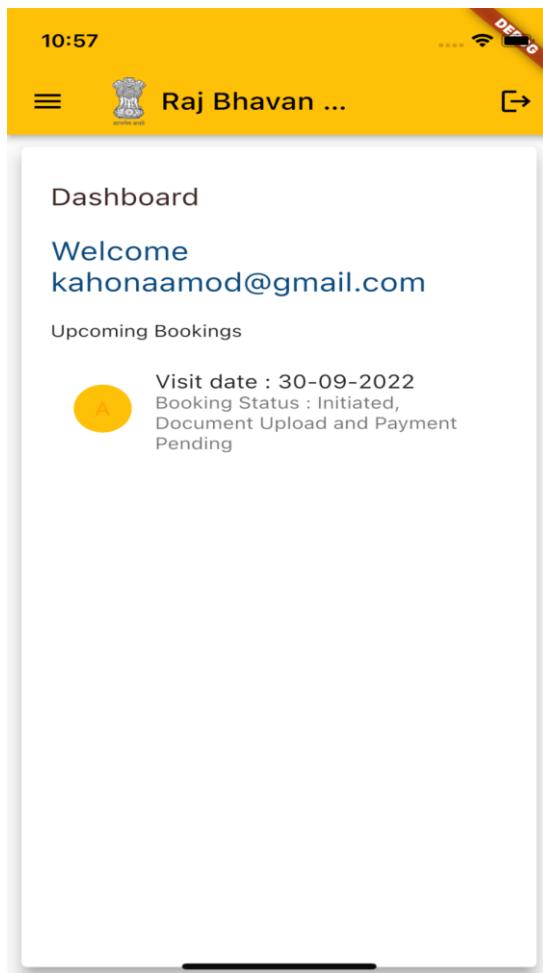
1) Login: After installation of the application, citizens can log in with Email Id/Mobile Number. The OTP will be send to Email Id / Mobile, after successful verification of OTP citizen can use the app.

After enter the email or mobile, click on the Next button , itwill ask for OTP which will receive by email/text message. If citizens didn't get OTP they can click on Don;t Receive OTP button to generate a new OTP after 60 seconds



Visitor Management System for Raj Bhavan (Mobile App)

2) Dashboard: with successful Login will show the citizen dashboard with the upcoming bookings (If booked a ticket) with the Login Name.



3) Book seats: To book a ticket for Raj Bhavan Citizens need to book a seat for visiting Raj Bhavan with their accompany detail.

The image displays two side-by-side screenshots of a mobile application interface for booking a visit to Raj Bhavan. Both screenshots show a yellow header bar with the Raj Bhavan logo and a menu icon. The top right corner of both screenshots has a red banner with the word "DEBUG".

**Screenshot 1 (Left): New Bookings**

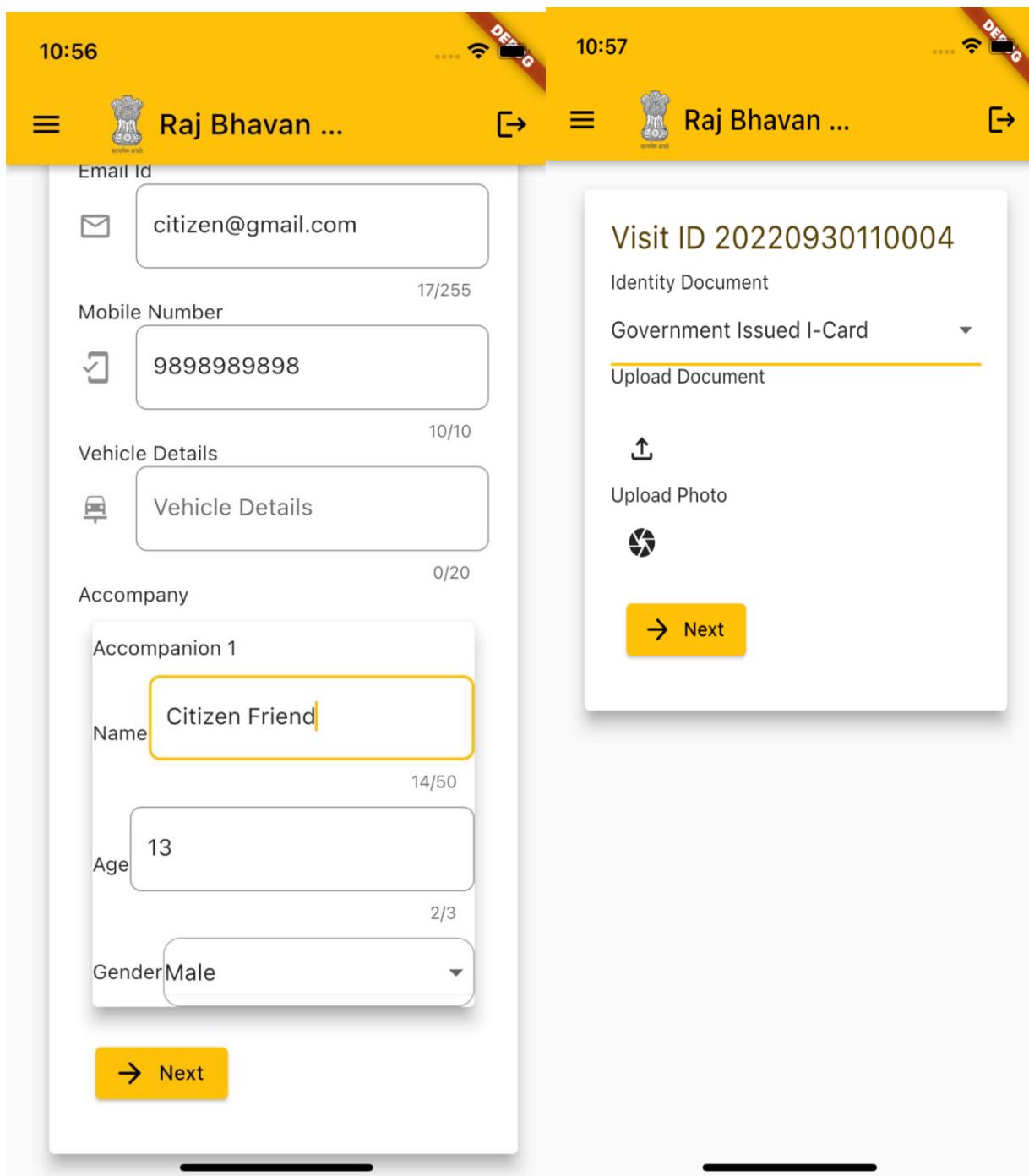
- Date: 2022-09-30
- Venue: Raj Bhavan Sunrise and Bunker
- Time Slot: 6:00AM - 9:00AM
- Number of Seats: 2
- Next button: A yellow button with a white arrow pointing right and the text "Next".

**Screenshot 2 (Right): Citizen Profile**

- Citizen: A placeholder text field with "Citizen" and a character limit of 8/50.
- Address: A placeholder text field with "Mumbai" and a character limit of 6/250.
- Date of Birth: 2009-01-01
- Gender:
  - Male: A selected radio button.
  - Female: An unselected radio button.
- Email Id: citizen@gmail.com
- Mobile Number: 9898989898
- Vehicle Details: Placeholder text field with "Vehicle Details" and a character limit of 10/10.
- Accompany: Placeholder text field with "Accompany" and a character limit of 0/20.

## Visitor Management System for Raj Bhavan (Mobile App)

After filling all the details, citizens have to click on the Next button and then upload the Required Documents.



After successful document uploading and payment, citizen will get the booking confirmation on screen, and by email.

## **CONCLUSION**

The System will enable administrators of Raj Bhavan to ease their visitors and visit management from a mobile Application. The Security Check of Visitors at the entry gate will be faster due to scanning the QR code. The Citizen will be benefitted by booking the visits easily from the mobile application.

## **Future Scope of the Project**

The project “**Mobile application for Raj Bhavan Visitor Management System**” can further elaborate on the following:

- 1) AI based visitor analysis can be implemented
- 2) Live location tracking of visitor based on geo location inside Raj Bhavan can be tracked
- 3) Integration of Payment Gateway

## **Experience Gained**

### **Summary of Experience Gained:**

The whole process of the development of the project has been an excellent chance for me to get exposure to Mobile application development and to learn new things every next day. Apart from the Language concepts during the course of the project, I got the chance to learn various aspects of flutter. Some of them are listed below: -

### **Technological aspects:**

- Handling Frontend development with API
- Developing and managing schedule

### **General aspects:**

- Designing the flow of the application
- Meeting deadlines
- Realtime documentation
- Testing on simulator

The knowledge gained and skills developed while working at NIC, Mumbai are sure to prove beneficial in the future. But the main thing is working on the live project is worth more than any other thing.

# Appendix

## DFD:

A **Data Flow Diagram (DFD)** is a graphic representation of the "flow" of data through business functions or processes. More generally, a data flow diagram is used for the visualization of data processing. It illustrates the processes, data stores, and external entities, data flows in a business or other system and the relationships between these things. Physical DFD's represent physical files and transactions, while logical or conceptual DFD's can be used to represent business functions or processes.

Data Flow Diagrams (DFDs) are a graphical representation of systems and systems components. They show the functional relationships of the values computed by a system, including input values, output values, and internal data stores. It's a graph showing the flow of data values from their sources in objects through processes/functions that transform them to their destinations in other objects. Some authors use a DFD to show control information, others might not. A DFD can be seen as a method of organizing data from its raw state.

## DFD NOTATIONS USED:

### **External Entities/Terminator**

These lie outside of the system being modelled. Terminators represent where the information comes from & where it goes. In designing a system, we have no idea about what these terminators do or how they do it.

### **Processes**

Modify the inputs in the process of generating the outputs.

### **Data Stores**

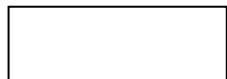
Represent a place in the process where data comes to rest. A DFD does not say anything about the relative timing of the processes, so a data store might be a place to accumulate data over a year for the annual accounting process.

### **Data Flows**

Represents how data moves between terminators, processes, & data stores (those that cross the system boundary are known as IO or Input Output Description).

### Symbols used for DFD:

An Open rectangle is used to represent a **Data Store**.



The rectangle is used to represent the **Data Source / Data Destination** within the system.



The arrow is used for showing the **Data Flow** in the system.



Ovals are used for showing the **Process** of the system.

### UML Diagram:

The Unified Modeling Language is a standard language for specifying, visualizing, constructing & documenting the artifacts of a software system, as well as for business modeling & another non-software system.

The UML design process involves creation of various graphical or text-based documents. In UML, these documents are called artifacts & they describe the outputs of a step in the process. The UML design process has two main parts:

**Analysis**-What the problem?

**Design**-How should the problem be solved?

The reason for this analysis & design process is to allow the project to be broken down into component parts, which provide the following project characteristics:

- Detail is hidden.
- The system is modular.
- Components are connected & interact.
- Layer complexity.
- Components may be reusable in other products.
- Variations on a theme.

A use case diagram is “a diagram that shows the relationships among actors & use cases within a system”. Use case diagram are often used to:

- Provide an overview of all parts of the usage requirements for a system or organization in the form of an essential model or a business model.
- Communicate the scope of a development project.

### Use case diagram would consist of:

#### **Use cases:**

A **use case** is a description of a system's behavior as it responds to a request that originates from outside of that system. The *use case* technique is used in software and systems engineering to capture the functional requirements of a system. Use cases describe the interaction between a primary actor—the initiator of the interaction—and the system itself, represented as a sequence of simple steps.

#### **Actors:**

Actors are something or someone which exists outside the system under study, and that take part in a sequence of activities in a dialogue with the system, to achieve some goal. They may be end users, other systems, or hardware devices. Each use case is a complete series of events, described from the point of view of the actor.

#### **Relationships:**

There are several types of relationship that may appear on a use case diagram:

- An association between an actor & a use case.
- An association between two use cases.

- A generalization between two actors.
- A generalization between two use cases.

### **System Boundary:**

The rectangle around the use cases is called the system boundary box & as the name suggests it indicates the scope of your system – the use case inside the rectangle represent the functionality that you intend to implement.

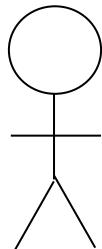
### Symbols used for Use Case diagram:



The ovals are used to show use case.

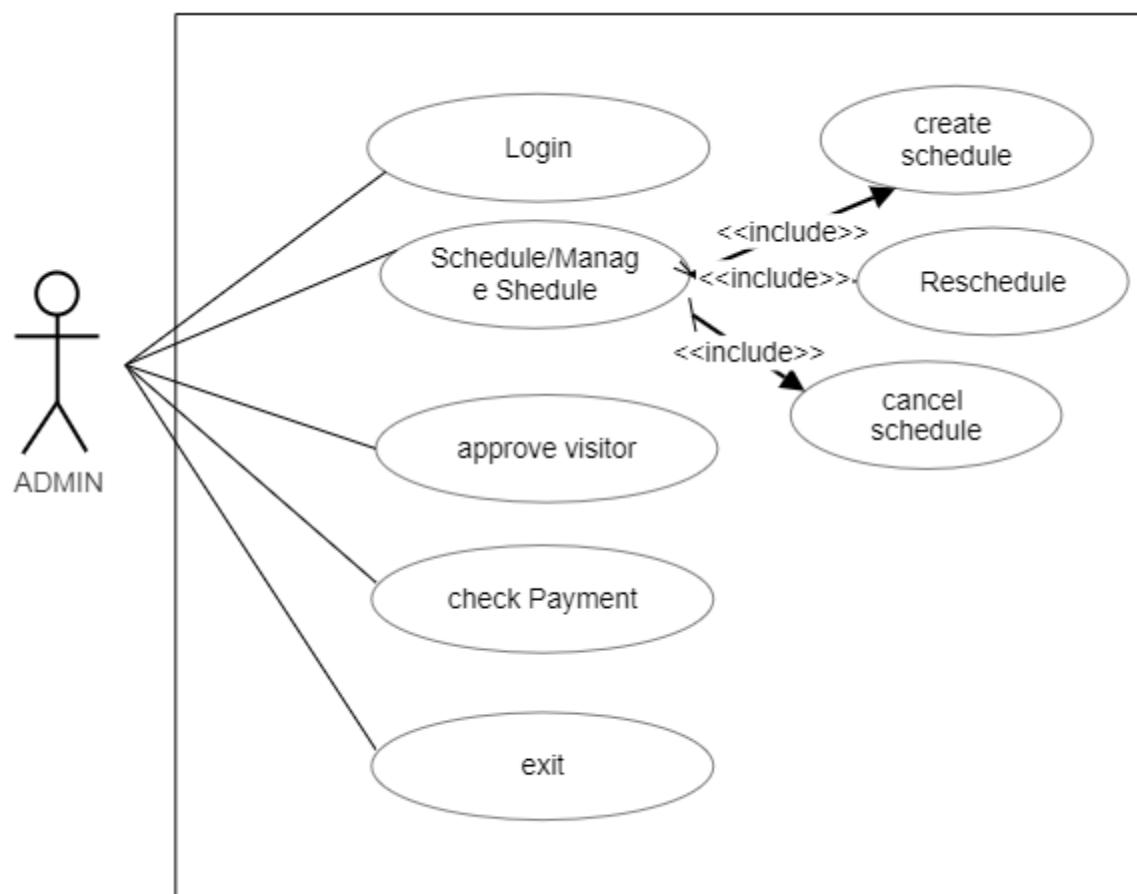


The rectangle is used to show system boundary.

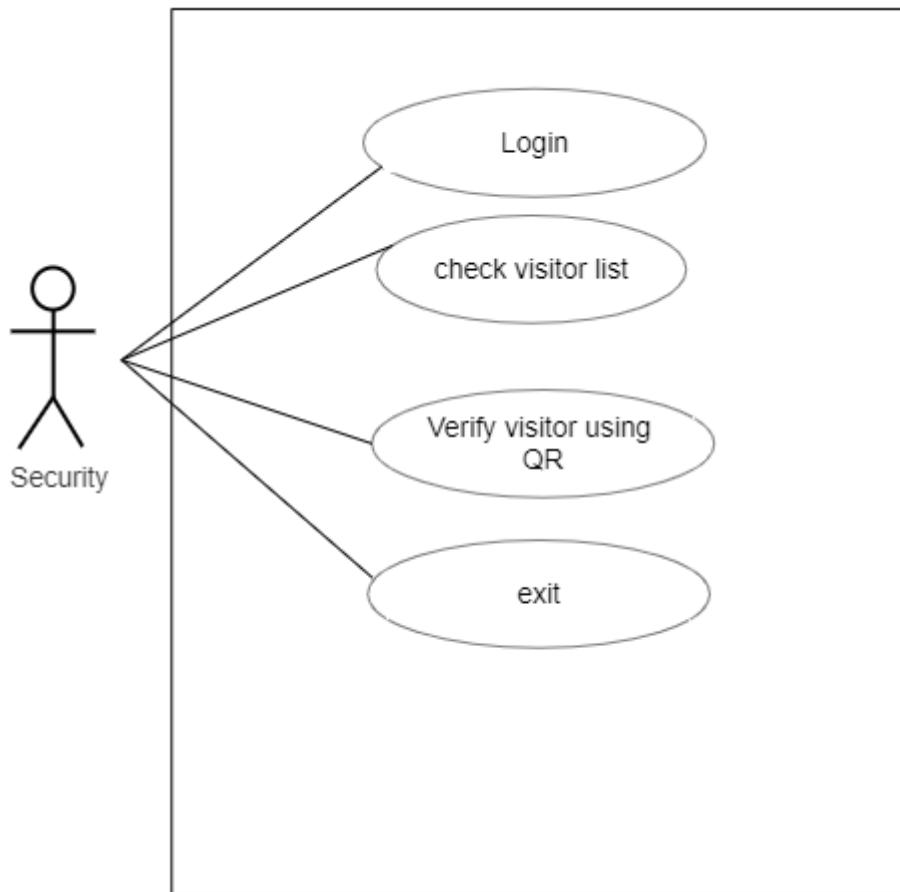


This is used to show an actor.

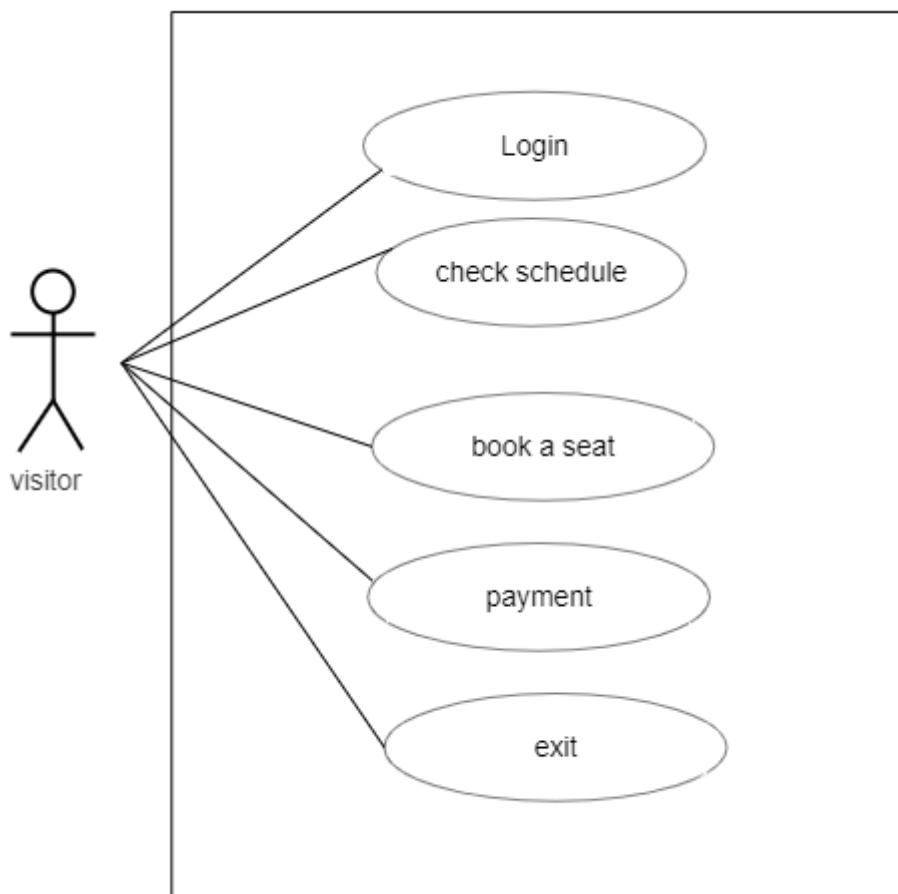
**Admin:**



## Security:



## Visitor:



## TESTING:

In a software project, errors can be injected at any stage during development. Because code is the only product that can be executed frequently & whose actual behavior can be observed, testing is the phase where the remaining errors from all the previous phases must be detected. Testing performs a critical role for quality assurance & for ensuring the reliability of software. During testing, the program to be tested is executed, with a set of test cases, & the output of the program for the test cases is evaluated to determine if the program is performing as expected.

Due to this approach, dynamic testing can only ascertain the presence of errors in the program; the exact nature of errors is not usually decided by testing.

Testing a large system is a complex activity, so for a project, incremental testing is generally performed, in which components & subsystems of the system are tested separately before integrating them to form the system for system testing.

**Testing** is a procedure to execute a program with the intent of finding errors. Testing is an important part of the software development process, to detect bugs, improve software design and to reduce usability problems. There are a number of test techniques such as the traditional unit, system, integration and acceptance tests, and a large number of software tools, which support testing in different ways. Testing flow proceeds from left to right and from top to bottom, user visible elements of the web application design are tested first followed by infrastructure design element.

## **Testing Objectives**

Testing is a process of executing a program with the intent of finding errors. A good test case is one that has a high probability of finding an as yet undiscovered error. A successful test is one that uncovers an undiscovered error.

## **Testing Principle**

- All tests should be traceable to the customer's requirements.
- Tests should be planned long before the testing begins.
- Testing should begin "in the small" & progress towards "the large".
- Exhaustive testing is not possible.

## ***Interface Testing / Unit Testing***

The overall strategy for interface testing is to:

- Uncover errors related to specific interface mechanisms. For e.g. errors in the proper execution of a menu link or the way the data is entered in the form.
- Uncover in the way the interface implements the semantics of navigation, web application functionality, or content display.
- To accomplish this strategy, a number of objectives must be achieved:
  - Interface features are tested to ensure that design rules, aesthetics and related visual content are available for the user without error.
  - Individual interface mechanisms are tested that are equivalent to unit testing.
  - Each interface is tested to ensure that it performs the requisite task.
  - The interface is tested within a variety of environments to ensure that it will be compatible.

## **Navigation Testing**

The job of this testing is:

- To ensure that the mechanisms that allow the web application user to travel through the web application are all functional and
- To validate that each navigation semantic unit can be achieved by the appropriate user category.

The first phase of navigation testing actually begins during Interface Testing. Navigation mechanisms are tested to ensure that each performs its intended function. Following navigation mechanisms should be tested:

- 1) ***Navigation links:*** Internal links within the web application, external links to other web application should be tested to ensure that proper content or functionality is reached when the link is chosen.

## **Integration Testing**

The purpose of unit testing is to determine that each independent module is correctly implemented. This gives little chance to determine that the interface between modules is also correct, and for this reason integration testing must be performed. One specific target of integration testing is the interface: whether parameters match on both sides as to type, permissible ranges, meaning and utilization.

The individual modules were aggregated one by one and functional testing was performed. The integrated product was considered as a black box and the output (report) is generated depending upon the data present in the flat files.

## **System Testing**

Software is to be incorporated with other system components (e.g. new hardware, information), and thus a series of special tests are to be conducted. Many times, software products are designed to run on a variety of hardware configurations. The software should actually be tested on much different hardware set – ups, although the full range of memory, processor, operating system, and peripheral possibilities may be too large for complete testing. There are many types of specifications and we should be aware of those as we perform system testing. During system testing, we should evaluate a number of attributes of the software that are vital to the user. These attributes represent the operational correctness of the product. Some of the attributes tested are:

- Usable:** The product is convenient and easy to use. A naïve user can also compare the flat files by browsing the files from their computer and selecting the corresponding interface. User can also add the new interface and can also make modifications in the existing interface.
- Secure:** Only the authorized user is able to use this tool.
- Compatible:** The product runs correctly using YUI\_EXT Library, works correctly in conjunction with existing software.

## **Performance Testing**

Performance testing is used to uncover performance problems that can result from lack of server-side resources, inappropriate network bandwidth, inadequate database capabilities, faulty or weak operating system capabilities, poorly designed web application functionality, and other hardware and software issues that can lead to degraded client-server performance.



## TEST PLAN

In Raj Bhavan mobile application we will use V-model software development life cycle. So, all the testing approaches will be followed. In the v-model approach, BRS/CRS/URS Is used for gathering information. After that, the document will be converted to SRS documents. After SRS document HLD and LLD models are designed.

The test cases will be written on the basis of different testing levels. As this system can be checked with the external environments, we may use the acceptance testing also to check Mobile applications.

1. Unit testing:
2. Integration testing
3. System testing

### Features:

1. User Roles: -
  1. Admin
  2. Security
2. Visitor List: Verifying that on searching all the visitor lists bookings satisfying the search criteria are visible on the visitor list screen.

### Test cases:

#### User Login:

U\_LOG\_1 user enters a wrong credentials display message username or password is incorrect.

U\_LOG\_2 user login enters correct username and password

U\_LOG\_3 user click on login button without entering credentials.

#### Visitor list:

VS\_1: Selects the date, venue, and time from the options that will display the date-wise list.

VS\_2: selects date-time from given options that will give an error message please select all fields.

<b>Project Name: Raj Bhavan Mobile Application</b>	
<b>Test Case Template</b>	
<b>Test Case ID:</b> Rajbhavan_admin1.0	<b>Test Designed by:</b> SHALINI
<b>Test Priority (Low/Medium/High):</b>	<b>Test Designed date:</b> 25/6/2022
<b>Module Name:</b> ADMIN	<b>Test Executed by:</b> SHRUTHISHRI
<b>Test Title:</b> To check the credentials of the login page.	<b>Test Execution date:</b> 27/6/2022
<b>Description:</b> This login page is an important gateway for accessing the contents. so the test case focus on checking the functionality of the login page.	
<b>Pre condition:</b> The login page should be in working condition and should open without error.	
<b>Dependencies:</b>	

## 5.2 TEST CASES

Test case ID	Steps to be executed	Test Data	Expected result	Actual Result	Pas s/Fai l	Remarks
	<b>For valid input</b>					
login 1.0.1	1. Enter User Name. 2. Enter Password. 3. Click on login button.	User="validusername"  Password="validpassword"	User should able to access the dashboard screen of the application without error.	User is able to access the dashboard screens of application without error.	Pass	--- //valid use case
login 1.0.2	1. Enter User Name. 2. Enter Password. 3. Click on login button.	User="Sec"  Password="124***"	User should able to access the next page of the application without error	User can get access to the next page of the application.	Pass	
	<b>For invalid input</b>					
login 1.0.3	1. Enter User Name. 2. Enter Password. 3. Click on login button.	User=" "  Password=" "	User should not enter in to the application	User is not able to enter in to the application	Pass	//invalid case use
login 1.0.4	1. Enter User Name. 2. Enter Password. 3. Click on login button.	User="ab "  Password="12 "	User should not enter in to the application	User is not able to enter in to the application	Pass	//invalid use case

Test case ID	Steps to be executed	Test Data	Expected result	Actual Result	Pas s/Fai l	Remarks
	<b>For valid input</b>					
login 1.0.1	1. visitor List. 2. Click on visitor list.	<b>Enter date="29/6/22"</b>  <b>Select venue="Raj bhavan sunrise and Bunker list"</b>  <b>Select schedule="6:00 AM – 9:00PM"</b>	User can access the visitor list detail	User can see all the details of visitor .	Pass	--- <b>//valid use case</b>
	<b>For invalid input</b>					
login 1.0.3	1. visitor List. 2. Click on visitor list.	<b>Enter date="29/6/22"</b>  <b>Select venue=""</b>  <b>Select schedule=""</b>	User can access the visitor list detail	User is not able to see the visitor list	Pass	<b>//invalid case use</b>
login 1.0.4	1. visitor List. 2. Click on visitor list.	<b>Enter date="29/6/22"</b>  <b>Select venue="Raj bhavan sunrise and Bunker list"</b>  <b>Select schedule=""</b>	User can access the visitor list detail	User is not able to see the visitor list	Pass	<b>//invalid use case</b>

# Bibliography

## **Bibliography**

1. <https://maharashtra.nic.in/>
2. <https://rajbhavan-maharashtra.gov.in/en/>