*Take One* movie Theatre Ticketing System

Requirments document for information development system

Project Phase Two

Johané le Roux 31614744

Christopher Slaghuis 31713858

Jean Marx 32313845

Reinardt Nel 27327884

James Uys 28461789

Contents

[**1.** **Project Plan** 2](#_Toc36651668)

[**2.** **Definitions, acronyms and abbreviations** 3](#_Toc36651669)

[**3.** **Project description and Scope** 4](#_Toc36651670)

[**4.** **Functional requirements** 6](#_Toc36651671)

[**5.** **Non-functional requirements (categorized according to the PIECES Framework)** 9](#_Toc36651672)

[**6.** **Candidate Systems Matrix** 10](#_Toc36651673)

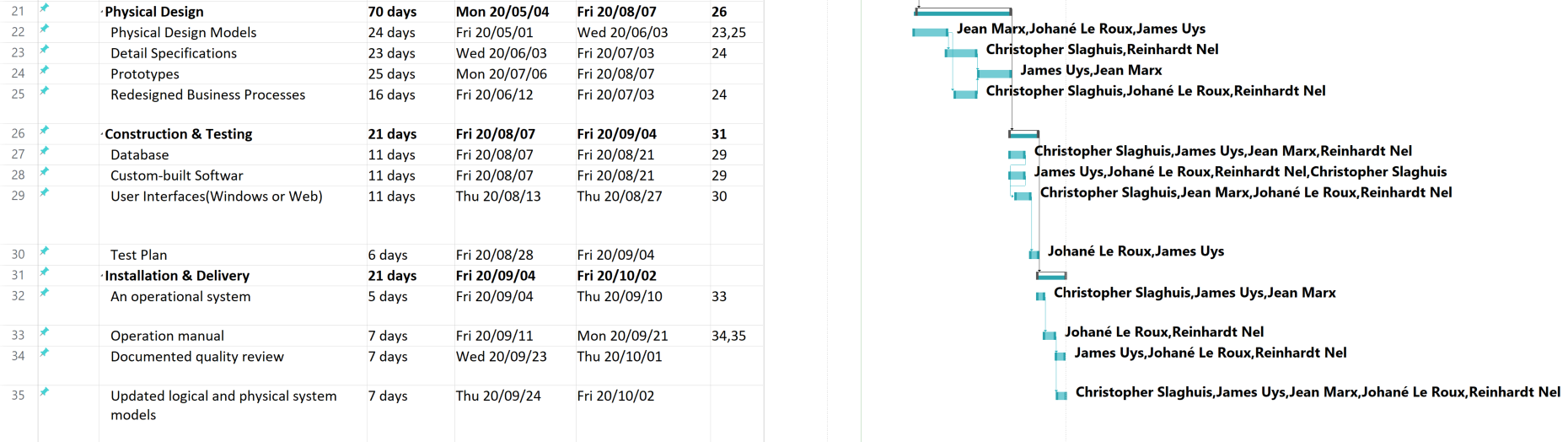
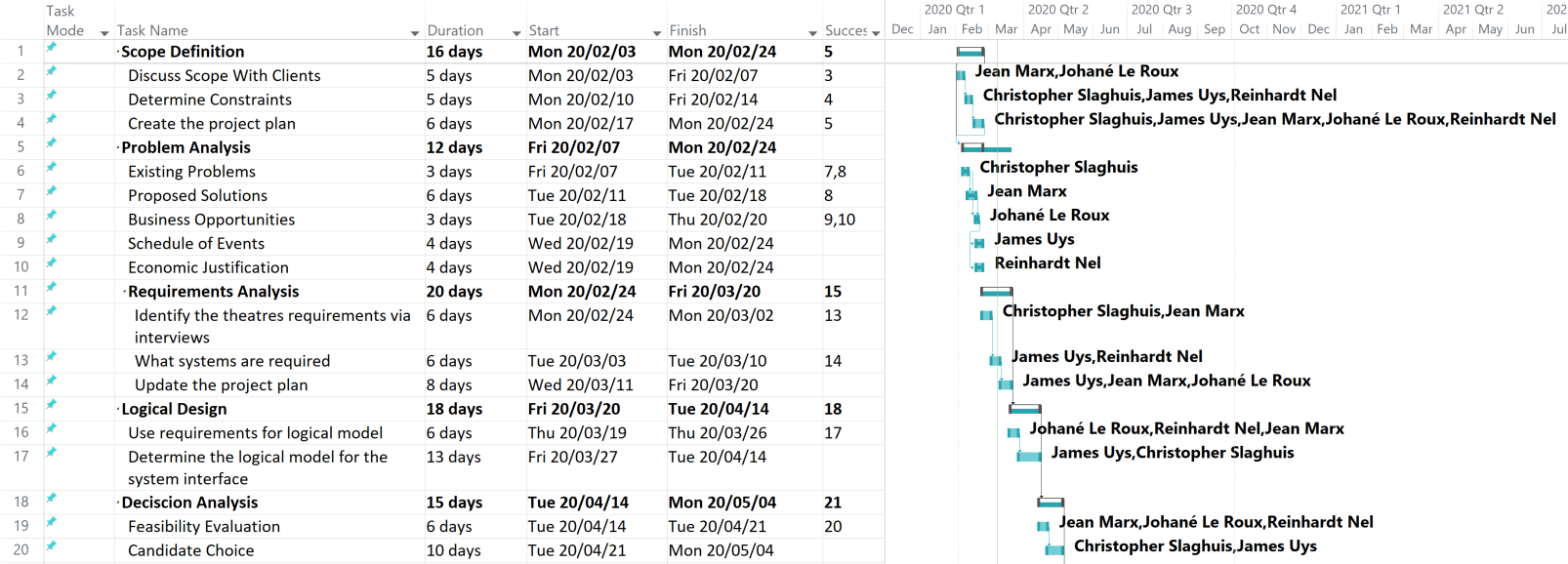
[**7.** **Feasibility Analysis Matrix** 12](#_Toc36651674)

[**8.** **Use-Case Glossary & Use-Case Model Diagram** 14](#_Toc36651675)

[**9.** **Examples of data, questionnaires, fact-finding techniques used** 16](#_Toc36651676)

[**10.** **Summary, future & further planning (PERT Chart)** 20](#_Toc36651677)

# **Project Plan**



# **Definitions, acronyms and abbreviations**

**Definitions:**

* FAST methodology: A hypothetical methodology used throughout this book to demonstrate a representative systems development process
* Fact-finding: The process of collecting information about system problems, opportunities, solution requirements, and priorities.
* PIECES Framework: A checklist for identifying problems with an existing information system.
* Functional requirement: A description of activities and services a system must provide.
  + inputs, outputs, processes, stored data
* Nonfunctional requirement: A description of other features, characteristics, and constraints that define a satisfactory system. (Qualities)
  + Performance, ease of learning and use, budgets, deadlines, documentation, security, internal auditing controls
* Use Case Diagram: A graphic depiction of the interactions among the elements of a system.
* [Use](https://searchsoftwarequality.techtarget.com/definition/use-case) Case: A methodology used in system analysis to identify, clarify, and organize system requirements.

**Acronyms and Abbreviations:**

* FAST: Framework for the Application of Systems Thinking
* PIECES: Performance, Information, Economics, Control, Efficiency of people and processes, Service to customers, suppliers, partners, employees.

# **Project description and Scope**

The main objective of this project is to provide an automated computerized system for the *Take One* movie theatre. The project must automate the daily business processes, provide extensive business reporting to support the business system and make tickets sales more technologically orientated. The system must provide extensive Help functionality to clients as well as employees using the system, backup functionality for future use and data must be protected against unwanted access to the system. Before the system can be used by clients and employees, all previous history data must be uploaded to the new system form the existing Excel system. The FAST methodology, which follows several phases, will be used to ensure a formal approach to the development and the implementation of the system. The following functionality will be included in the scope of this project.

The system must include functionality for the following:

1. Maintain clients
2. Maintain movie schedule
3. Maintain genres
4. Maintain special occasions
5. Maintain movies
6. Book seats online
   1. Notify client about ticket information when sale is complete.
   2. The notification will include their movie ticket code.
   3. Show seat that are booked.
7. Selling tickets (at the movie theatre)
8. Export data
9. Request report:

* List of movies per time period
* Top 10 movie genres per time period (e.g. comedy, drama, action)
* Top 10 movies per time period (time of day, day of weekend)
* Top 10 movies per schedule

In addition, the system must satisfy the following non-functional requirements:

* Extensive Help function for clients as well as for employees
* Provide unique identifiers for employees using the system to ensure secure and authorized access to the system
* The current database will hold the data for the movies that are showing at the theatre (all data and information regarding the movies), movies that was shown per time period (not currently) and special occasions that was hosted by the movie theatre.
* Queries performed by the employees should take no longer than 3 seconds.
* The system will be accessible by two types of users. The authorized employees will have access to all functionality of the system and will be able to change the database as they wish. The second type of users will be the client using the system to book tickets for movies either over the Internet or at the counter at the movie theatre. The clients will not have access to databases or the functionality of the system.

# **Functional requirements**

Login Form

Maintain

Genres

Movie Schedule

Book Seats

Special occasions

Sell tickets

Movies

Backup Data

Reporting

|  |  |  |
| --- | --- | --- |
| **Maintain** | | |
| **Add new movies** | | |
| Input Data | Processing | Output |
| * Movie description * Movie genre * Ticket Price * Movie length * Cast and directors’ names | * Display a warning message if movie to be added is similar to a currently added movie. * When the save button is clicked and no movies of the same title exist, save the record to the database as well as mark movie times on the schedule. * Make the movie able to be booked by the public | * Display a warning message if a movie with the same title already exists. * Display a new message when the new movie has been added to the database and added to the schedule. * Display the scheduled movie time in the schedule. |

|  |  |  |
| --- | --- | --- |
| **Maintain** | | |
| **Edit Movies** | | |
| Input Data | Processing | Output |
| * Select Movie to edit * Select field to edit * Make edits to field * Click the edit button | * Display a warning message before editing * When the edit button is clicked validate all new data entered * Make the edited movie able to be booked by the public | * Display a warning message before editing * Display a new message when the edited movie has been added to the database and added to the schedule. * Display the scheduled movie time in the schedule. |

|  |  |  |
| --- | --- | --- |
| **Maintain** | | |
| **Schedule movies (ADD, EDIT, DELETE)** | | |
| Input Data | Processing | Output |
| * Select movie to schedule * Select time slot | * When the save button is clicked and no movies clash, save the record to the schedule. * Make the movie able to be booked by the public. * Check that there are no clashes between scheduled movies. * Display a warning message if a scheduled time is booked by a different movie. * When editing and deleting check to make sure no one has book the previously scheduled movie. | * Display a warning message if a scheduled time is booked by a different movie. * Display a message when the movie has been added to the schedule. * Display the scheduled movie time in the schedule. |

|  |  |  |
| --- | --- | --- |
| **Book tickets** | | |
|  | | |
| Input Data | Processing | Output |
| * Select a movie to book * Select a time slot * Select a seat | * Verify that the seats are available to be booked. * Display a warning message if the selected seats are already booked * Save the seat selection the database * Update the available seats for that movie | * Display a warning message if the selected seats are already booked * Display a message when the seats have been booked * Sent tickets to the customer. |

|  |  |  |
| --- | --- | --- |
| **Backup Data** | | |
|  | | |
| Input Data | Processing | Output |
| * Click a Button to backup data | * Check to see if there is enough available space in backup location. * Close the movie database * Copy database to backup location * Open database * Recorded date at time of backup | * Display a warning message if there is not enough space available in backup location * Display message when backup is complete * Display message if an error occurs |

# **Non-functional requirements (categorized according to the PIECES Framework)**

|  |  |
| --- | --- |
| **Requirements** | **Piece categories** |
| Have unique identifiers for all employees and management | Security |
| Have a through help function | Efficiency of People and Processes |
| The database will hold information of all movies ever showed as well as when they were shown. | Performance  Information and Data |
| All queries made to the database should never take more than 5 Seconds | Performance |
| There will be three different users of the system. Users that will book tickets, employees who will be able to amend bookings and management who will be able to amend bookings, schedule movies and view reports created by the program. | Control or Security |

# **Candidate Systems Matrix**

|  |  |  |  |
| --- | --- | --- | --- |
| **Characteristics:** | **Candidate 1:** | **Candidate 2:** | **Candidate 3:** |
| Portion of System Computerized: | N/A. | Tickets will be sold electronically, online, as well as the bookings for seats. Records of data will also be stored electronically. | Same as candidate 2. |
| Benefits: | No costs related to maintaining hard and -software. | Quicker, more efficient method to perform transactions, bookings and recordings. Cost of physical books and tickets are reduced. New released would be easier to promote and advertise. | Same as candidate 2. |
| Additional Software: | N/A. | Any cloud-storage service to keep track of records, sales and bookings. | A server must be purchased where data can be stored. |
| Servers and workstation: | A room or location where physical records (books with the data) can be stored. As well as a station for employees to physically sell tickets and record seat bookings. | No workstation other than refreshments and scanning of booked tickets are required, with the transactions and bookings being done online. No servers required. | Same as candidate 2. But with servers to store the necessary date on. |
| Software Tools Needed: | N/A. | Visual Studio 2019  Internet Explorer | Same as candidate 2.  Microsoft Server |
| Method of Data Processing: | The sale of tickets and booking of seats are all done physically (books, hard copies) | Tickets are bought online, and a seat is booked. Data is exported to the application to verify the tickets. Data is then sent to cloud-storage | Same as candidate 2.  Except data is stored on the server instead. |
| Application Software: | N/A. | Custom solution. | Same as candidate 2. |
| Output Devices and Implications: | A printer to print tickets. | N/A. | Same as candidate 2. |
| Input Devices and Implications: | N/A. | Scanner to verify the code the customer received. | Same as candidate 2. |
| Storage Devices and Implications: | Only physical, hard copies of the records of sales and bookings. | Cloud-storage software. The age range of customers and what movies and genres are most popular in these ranges. What days are the busiest. The profit/loss margin for a certain financial period. | Same as candidate 2.  Instead data is stored on a server. |
| Output Software: | N/A. | N/A. | N/A. |

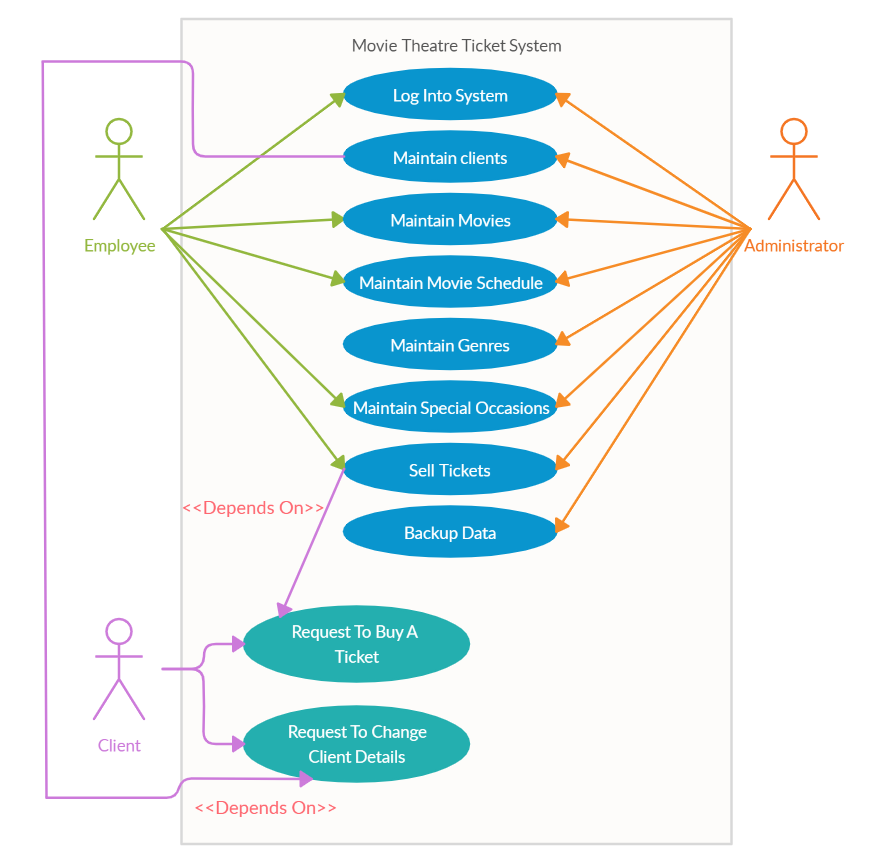
# **Feasibility Analysis Matrix**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feasibility Criteria:** | **Weight:** | **Candidate 1** | **Candidate 2** | **Candidate 3** |
| Operational Feasibility: | 10 | The system in place is working but requires a lot of physical items in order to continue operations. Could be more user friendly and effective. | Would support all requirements specified, would also improve efficiency as well as save costs. | Same as candidate 2. |
|  |  | **Score: 60** | **Score: 95** | **Score:95** |
| Cultural Feasibility: | 10 | The users often complain about waiting times and the lack of flow. | The users would be happy with this change, as they can book tickets and seats from the comfort of their home and simply show up with their ticket-code. | Same as candidate 2. |
|  |  | **Score:70** | **Score:90** | **Score:90** |
| Technical Feasibility: | 20 | The system in place requires a lot of physical-related materials in order to function. Which in the long run could cause storage and be too disorganized. | The system would be easy to work with and maintain. Would provide the adequate number of features. | Same as candidate 2.  Except the cost of buying, maintaining and using a server would increase. |
|  |  | **Score:70** | **Score:95** | **Score:85** |
| Risk Feasibility: | 20 | The only real risk involved would be in the case a fire breaks out and destroys all records of data. | Minimal risk. Only in the case of malicious attacks, power outages or gross mismanagement would the system not function as intended. Backups of data are easy to make. | Same as candidate 2.  With the added risk of system shutdown in the case(s) of disaster. |
|  |  | **Score:85** | **Score:80** | **Score:70** |
| Economic Feasibility: | 20 | The repeated purchase and printing of tickets will be a constant cost. | The system would not be expensive to design and maintain. It would also remove the cost of physical materials. | Same as candidate 2.  The addition of a server would require more maintenance (the appointment of a database manager) and technicalities. |
|  |  | **Score:65** | **Score:90** | **Score:80** |
| Schedule Feasibility: | 10 | N/A. | Due to the non-complex nature of the system, if proper scheduling is done and executed, there ought to be no problems | Same as candidate 2.  The server would require more time to implement and integrate into the system. Which would require a change in scheduling in order to accommodate. |
|  |  | **Score:100** | **Score:95** | **Score:85** |
| Legal Feasibility: | 10 | N/A. | N/A. | N/A. |
|  |  | **Score:100** | **Score:100** | **Score:100** |
| **Weighted Score:** | 100 | **Score:78** | **Score:91** | **Score:86** |

# **Use-Case Glossary & Use-Case Model Diagram**

**Use-case glossary**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | **Use Case Description** | **Participating Actors and Roles** |
| **Log into system** | The process a user performs to get access to the system by providing needed credentials | Administrator(s) Employee |
| **Maintain Client** | The event where a new client will be added to the system, the details of an existing client changes or an existing client is removed | Administrator(s) |
| **Maintain Movie** | The event where a new movie will be added to the system, the details of an existing movie changes or an existing movie is removed | Administrator(s)  Employee |
| **Maintain Movie Schedule** | The event where a new schedule will be added to the system, the details of an existing schedule changes or an existing schedule is removed | Administrator(s) Employee |
| **Maintain Genres** | The event where a new genre will be added to the system, the details of an existing genre changes or an existing genre is removed | Administrator(s) |
| **Maintain Special Occasions** | The even where a new special occasion will be added to the system, the details of an existing special occasion changes or an existing special occasion is removed | Administrator(s) Employee |
| **Sell Tickets** | This use case describes the event in which a ticket is sold to a client | Administrator(s) Employee |
| **Backup Data** | This use case describes the event where the database is backed up to the cloud | Administrator(s) |

**Use-case model**

# **Examples of data, questionnaires, fact-finding techniques used**

**User requirements and system problems interview**

Interviewee: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Occupation: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_/\_\_\_\_/\_\_\_\_

* Introduction and reason for interview.
* What system is currently in place at the theatre? How does it keep track of bookings and tickets?
* Would an automated system improve this process? If so, how?
* What are some of the biggest requirements the theatre would require of the system?
* By what date would you like the system to be completed and implemented? At latest?

**Questionnaire**

Name & Surname: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_/\_\_\_\_/\_\_\_\_

* Is the current system automated?

□ Yes □ No

* Do you understand the current information system well?

□ 1 □ 2 □ 3 □ 4 □ 5

* Does the current system allow you to perform your job well, or could there be improvements?

□ Performs well □ Can improve

* Is the current system reliable if so, on what scale?

□ Yes □ No

□ 1 □ 2 □ 3 □ 4 □ 5

* Would implementing a new system ease your job?

□ Yes □ No

* Does the current system support online booking?

□ Yes □ No

* What new features would you like to be implemented/improved:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Summary of findings**

* Is the current system automated?

|  |  |
| --- | --- |
| **Answer:** | **Count** |
| Yes | 3 |
| No | 2 |

* Do you understand the current information system well?

|  |  |
| --- | --- |
| **Answer:** | **Count** |
| 1 | 0 |
| 2 | 1 |
| 3 | 2 |
| 4 | 1 |
| 5 | 1 |

* Does the current system allow you to perform your job well, or could there be improvements?

|  |  |
| --- | --- |
| **Answer:** | **Count** |
| Performs well | 3 |
| Can improve | 2 |

* Is the current system reliable if so, on what scale?

|  |  |
| --- | --- |
| **Answer:** | **Count** |
| Yes | 3 |
| 1 | 0 |
| 2 | 1 |
| 3 | 0 |
| 4 | 1 |
| 5 | 1 |
| No | 2 |

* Would implementing a new system ease your job?

|  |  |
| --- | --- |
| **Answer:** | **Count** |
| Yes | 2 |
| No | 3 |

* Does the current system support online booking?

|  |  |
| --- | --- |
| **Answer:** | **Count** |
| Yes | 0 |
| No | 5 |

* What new features would you like to be implemented/improved:
  + Allowing a QR with the ticket information to be sent to the customer via WhatsApp, much like Howler’s system.
  + N/A
  + N/A
  + The user friendliness could be much improved, especially for new users.
  + N/A
  + Reporting for management could be much improved, especially with historical data.

**Summary of findings graphs**

# **Summary, future & further planning (PERT Chart)**

This concludes the requirements document for the Take One movie theatre’s ticketing system. All relevant components regarding the Requirement Analysis Phase has been completed, as well as the Scope Definition and the Problem Analysis which was done earlier. The project team will engage the Logical Design phase of the project – the end date of which is assigned to 04/05/20 – pending the acceptance of this Requirements document.