Wireless Sensor-based Application for Managing Theme Park

Software

Requirements specification

**By**

Pabhawee Chuacharoen 552115037

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

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

Dr.Noppon Choosri

**Document History**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Version | Status | Change Context | Viewable | Reviewer | Responsible | Date |
| V.1.0 | Draft | Add Introduction  Overall | PC, NC | PC | PC | 08/08/15 |
| V.1.1 | Draft | Add Requirements | PC, NC | PC | PC | 10/08/15 |
| V.1.2 | Draft | Add Use case diagram  Add Activity diagram  Add Use case description | PC, NC | PC | PC | 11/08/15 |
| V.1.3 | Draft | Fix Use case description Add new requirement | PC, NC | PC, NC | PC | 12/08/15 |
| V.2.0 | Release | Fix activity diagram  Change project name | PC, NC | PC, NC | PC | 26/08/15 |
| V.3.0 | Release | New requirements | PC, NC | PC, NC | PC | 22/10/15 |

PC = Pabhawee Chuacharoen , NC = Dr. Noppon Choosri

**Table of Content**

[Chapter One | Introduction 3](#_Toc434787716)

[1.1. Purpose 3](#_Toc434787717)

[1.2. Software Scope 3](#_Toc434787718)

[1.3. Operation Environment 3](#_Toc434787719)

[1.4. Definition, Acronyms and Abbreviation 3](#_Toc434787720)

[1.4.1 Key Definition 3](#_Toc434787721)

[1.4.2 Key Acronyms and Abbreviation 4](#_Toc434787722)

[Chapter Two | Overall Description 5](#_Toc434787723)

[2.1. Product Perspective 5](#_Toc434787724)

[2.2. User Characteristics 5](#_Toc434787725)

[2.3. Features 5](#_Toc434787726)

[Chapter Three | Requirement 6](#_Toc434787727)

[3.1. User Requirements 6](#_Toc434787728)

[Chapter Four | Specific Requirement 9](#_Toc434787729)

[4.1. Use Case Diagram 9](#_Toc434787730)

[ Wireless sensor-based application for managing theme park overview 9](#_Toc434787731)

[ Component of Wireless Sensor-based Application for Managing Theme Park 9](#_Toc434787732)

[4.2. Use Case Description 11](#_Toc434787733)

[2.1. UC-1 Register guest into the system 12](#_Toc434787734)

[2.2. UC-1 View guest information from the system 14](#_Toc434787735)

[2.3. UC-3 Edit guest information from the system 15](#_Toc434787736)

[2.4. UC-4 Delete guest information from the system 17](#_Toc434787737)

[2.5. UC-5 View attraction information from the system 18](#_Toc434787738)

[2.6. UC-6 Add attraction information into the system 18](#_Toc434787739)

[2.7. UC-7 Edit attraction information from the system 21](#_Toc434787740)

[2.8. UC-8 Delete attraction information from the system 22](#_Toc434787741)

[2.9. UC-9 Enter the entrance gate by representing RFID tag at RFID reader 23](#_Toc434787742)

[2.10. UC-10 Exit the exit gate by representing RFID tag at RFID reader 24](#_Toc434787743)

[2.11. UC-11 Enter attraction entrance by representing RFID tag at RFID reader 25](#_Toc434787744)

[2.12. UC-12 View a report on numbers of guest visiting the theme park within one week range. 26](#_Toc434787745)

[2.13. UC-13 View a report on numbers of guest attending each ride within one week range 27](#_Toc434787746)

# Chapter One | Introduction

## Purpose

The purpose of this Software Requirement Specification (SRS) is to provide the description of the functional requirement for the project. The requirements explain the steps to use the system to encourage the users and developers to communicate and to be straight in the detail of each requirement. In addition, the SRS document is the part of the software design phase, which is leading to make a consistency between software product and functional requirement.

## Software Scope

The system designed by the requirement of user. The feature was divided for three main users, which are the administration of the system, ticket seller and the guest of the theme park. The system designed in order to manage the information of the guest of the park and manage the attraction information. In each feature of the system are differently use by the user of the system. The listed of the main feature was wrote in the project plan document in the project overview topic.

## Operation Environment

The wireless sensor-based application develops in the form of web-based application. The users need to connect to internet in order to access to this application and the devices that are attached to guests/attractions must be in range of wireless network.

## Definition, Acronyms and Abbreviation

## Key Definition

| Name | Explain |
| --- | --- |
| IEEE | Institute for Electrical and Electronics Engineers. Biggest global interest group for engineers of different branches and for computer scientists. [IEEE90] |
| Requirement | (1) A condition or capability needed by a user to solve a problem or achieve an objective. (2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. (3) A documented representation of a condition or capability as in definition (1) or (2). [IEEE90] |
| Feature | Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of a product in the language of the product. Used for requirements analysis, design, coding, testing or maintenance. [IEEE90] |
| Use Case | (1) Concept to describe a system based on usage of system resources by its environment. Characterized by an objective set of interactions within and at the borders of that system.  (2) Notation from UML for describing a scenario (Usage approach, operational scenario) from the perspective of tis users. [IEEE90] |
| Station | The station for the guest of the park in order to make the time reservation. |
| Module | The Arduino, which programmed to be the RFID reader. |
| Software Engineering | The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. [IEEE90] |
| Specification | Precise description of an activity or work product which serves as basis or input for further activities or work product. A specification can comprise requirements to a product and how they will be solved. Different parts of a specification (e.g., what is to be done, how it will be done) must not be mixed. [IEEE90] |
| Guest Information | The information of guest including name, e-mail and mobile number. |
| Attraction Information | The information of attraction including name, description and height limit. |

## Key Acronyms and Abbreviation

|  |  |
| --- | --- |
| Name | Explain |
| WSAT | Wireless sensor-based application for managing theme park |
| GIMS | Guest information management system |
| AIMS | Attraction information management system |
| SRS | Software Requirement Specification |
| URS | User Requirement Specification |
| UC | Use Case Description |
| UI | User Interface |
| UID | User Identification |
| RFID | Radio-frequency identification |

# Chapter Two | Overall Description

## Product Perspective

The WSAT is aimed to prove concept of using wireless sensor technology for theme park management with The RFID tag can store guest's UID and communicate by using the RFID reader to collect.

## User Characteristics

The users of the system are divided by the different role of the user in the theme park. There are ticket seller, administration of the system and the guest of the theme park.

* Ticket seller who manage guest information
* Administration who manage attraction information
* Guest who intended to pay for the ticket and access to services within the park using RFID tag.

## Features

Feature#1 Guest information management

Feature#2 Attraction information management

Feature#3 Collecting visiting statistics data for attraction information

Feature#4 Attractions' statistic information

# Chapter Three | Requirement

## User Requirements

**URS-01: Ticket sellers can register guest into the system.**

**SRS-01:** System shall get UID to the system.

**SRS-02:** System shall provide a user interface for the ticket seller to fill in guest’s registration form.

**SRS-03:** System shall validate the guest’s information, which are name, e-mail and phone number.

**SRS-04:** System shall provide a submit button.

**SRS-05:** System shall provide the error message “Please fill out this field” when the guest name field is empty.

**SRS-06:** System shall provide the error message “Please fill out this field” when the email field is empty.

**SRS-07:** System shall provide the error message “Please fill out this field "when the mobile no. field is empty.

**SRS-08:** System shall provide the error message “Name must be 5-30 characters in length "when length of the guest name field is exceed 30 or less than 5 characters.

**SRS-09:** System shall provide the error message “Invalid mobile no." when length of the mobile no. field is not 10.

**SRS-10:** System shall provide the error message “Please enter an email address" when the input in the email field is incorrect format.

**SRS-11:** System shall add the guest’s information into the system.

**URS-02: Ticket seller can view guest information from the system.**

**SRS-12:** System shall provide a table to display guest’s information.

**URS-03: Ticket seller can edit guest information from the system.**

**SRS-13:** System shall provide the edit button for each guest.

**SRS-14:** System shall provide the form for ticket seller to edit the information.

**SRS-04:** System shall provide a submit button.

**SRS-05:** System shall provide the error message “Please fill out this field” when the guest name field is empty.

**SRS-06:** System shall provide the error message “Please fill out this field” when the email field is empty.

**SRS-07:** System shall provide the error message “Please fill out this field "when the mobile no. field is empty.

**SRS-08:** System shall provide the error message “Name must be 5-30 characters in length "when length of the guest name field is exceed 30 or less than 5 characters.

**SRS-09:** System shall provide the error message “Invalid mobile no." when length of the mobile no. field is not 10.

**SRS-10:** System shall provide the error message “Please enter an email address" when the input in the email field is incorrect format.

**SRS-15:** System shall save the updated guest’s information into the system.

**URS-04: Ticket seller can delete guest information from the system.**

**SRS-16:** System shall provide the delete button.

**SRS-17:** System shall provide the confirm dialog box with the message “Are you sure you want to delete this record?”.

**SRS-18:** System shall delete the guest’s information from the system.

**URS-05: Admin can view attraction information from the system.**

**SRS-19:** System shall provide a table to display attraction’s information.

**URS-06: Admin can add attraction information into the system.**

**SRS-20:** System shall provide a user interface for the admin to fill in attraction’s registration form.

**SRS-21:** System shall validate the name, description and height limit field.

**SRS-22:** System shall provide the error message “Please fill out this field” when the attraction name field is empty.

**SRS-23:** System shall provide the error message “Please fill out this field” when the description field is empty.

**SRS-24:** System shall provide the error message “Please fill out this field” when the height limit field is empty.

**SRS-25:** System shall provide the error message “Name must be 5-30 characters in length "when length of the attraction name field is exceeding 30 or less than 5 characters.

**SRS-26:** System shall provide the error message “Description must be 5-60 characters in length" when length of the description field is exceeding 60 or less than 5 characters.

**SRS-27:** System shall provide the error message “Please enter the number" when the input in the height limit field is not numeric.

**SRS-28:** System shall provide the error message “Height limit must not be more than 3 digits “when the input in the height limit field is exceed 3 digits.

**SRS-04:** System shall provide a submit button.

**SRS-29:** System shall add the inputted attraction’s information into the system

**URS-07: Admin can edit attraction information from the system.**

**SRS-30:** System shall provide the edit button for each attraction.

**SRS-31:** System shall provide the form for admin to edit the attraction information.

**SRS-21:** System shall validate the name, description and height limit field.

**SRS-22:** System shall provide the error message “Please fill out this field” when the attraction name field is empty.

**SRS-23:** System shall provide the error message “Please fill out this field” when the description field is empty.

**SRS-24:** System shall provide the error message “Please fill out this field” when the height limit field is empty.

**SRS-25:** System shall provide the error message “Name must be 5-30 characters in length "when length of the attraction name field is exceed 30 or less than 5 characters.

**SRS-26:** System shall provide the error message “Description must be 5-60 characters in length" when length of the description field is exceeding 60 or less than 5 characters.

**SRS-27:** System shall provide the error message “Please enter the number" when the input in the height limit field is not numeric.

**SRS-28:** System shall provide the error message “Height limit must not be more than 3 digits “when the input in the height limit field is exceed 3 digits.

**SRS-04:** System shall provide a submit button.

**SRS-32:** System shall save the updated information into the system.

**URS-08: Admin can delete attraction information from the system.**

**SRS-16:** System shall provide the delete button.

**SRS-17:** System shall provide the confirm dialog box with the message “Are you sure you want to delete this record?" .

**SRS-33:** System shall delete the attraction's information into the system.

**URS-09: Guest can enter the entrance gate by presenting RFID tag to the RFID reader.**

**SRS-34:** System shall provide the RFID reader monitoring interface to indicate that the RFID is connected to the application.

**SRS-35:** System shall read the UID from the guest's LF-RFID tag that belong to the card holder.

**SRS-36:** System shall send the UID and location where the arduino is deployed from the arduino device to the web application.

**SRS-37:** System shall store the UID, datetime and the location into the database.

**URS-10: Guest can exit the exit gate by presenting RFID tag to the RFID reader.**

**SRS-34:** System shall provide the RFID reader monitoring interface to indicate that the RFID is connected to the application.

**SRS-35:** System shall read the UID from the guest's LF-RFID tag that belong to the card holder.

**SRS-36:** System shall send the UID and location where the arduino is deployed from the arduino device to the web application.

**SRS-37:** System shall store the UID, datetime and the location into the database.

**URS-11: Guest can get through attraction entrance by representing RFID tag at RFID reader.**

**SRS-34:** System shall provide the RFID reader monitoring interface to indicate that the RFID is connected to the application.

**SRS-35:** System shall read the UID from the guest's LF-RFID tag that belong to the card holder.

**SRS-36:** System shall send the UID and location where the arduino is deployed from the arduino device to the web application.

**SRS-37:** System shall store the UID, datetime and the location into the database.

**URS-12: Admin can view a report on numbers of guest visiting the theme park within one week range.**

**SRS-38:** System shall provide a line chart to display the number of visitors on current day and the last 6 days.

**SRS-39:** System shall display the message "The activity record is empty" if there are no record of the selected location.

**URS-13: Admin can view a report on numbers of guest attending each ride within one week range.**

**SRS-40:** System shall provide the user interface for users to select an attraction to view the activity.

**SRS-38:** System shall provide a line chart to display the number of visitors on current day and the last 6 days.

**SRS-39:** System shall display the message "The activity record is empty" if there are no record of the selected location.

# Chapter Four | Specific Requirement

## Use Case Diagram

### Wireless sensor-based application for managing theme park overview

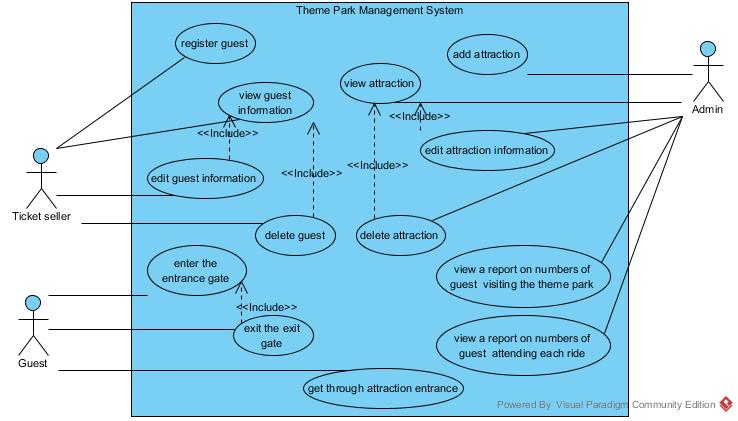


Figure-1 : Use case diagram to show Theme Park Management System by Using RFID overview

### Component of Wireless Sensor-based Application for Managing Theme Park

#### Guest Information Management System

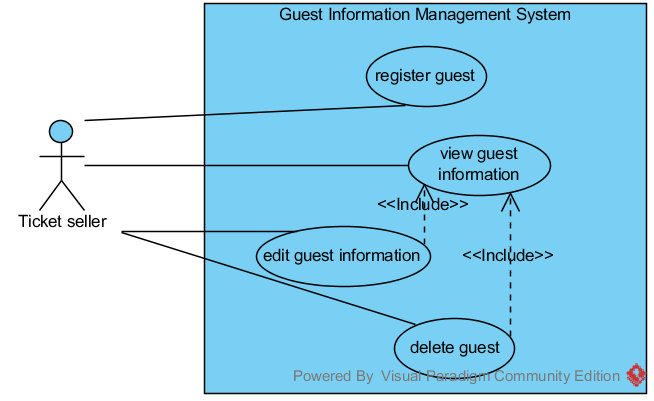


Figure-2 : Use case diagram to show guest information management system overview

* **Guest information management system**

The component deals with the guest information management tasks. The Ticket Seller first can registers guest information into the system, after that ticket seller can edit and delete guest information as well as view the guest information.

#### Attraction information Management System

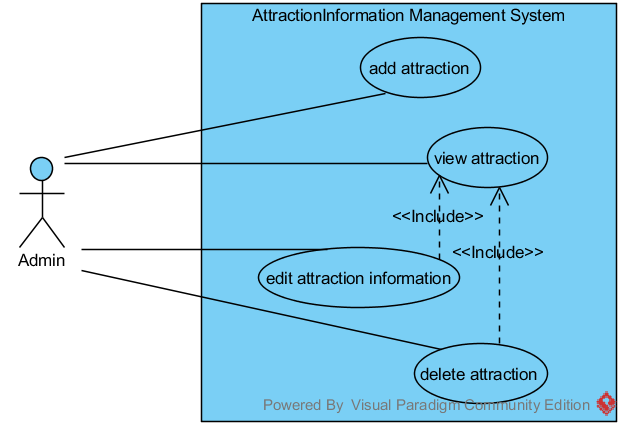


Figure-3 : Use case diagram to show attraction information management system overview

* **Attraction Information Management System**

The component deals with the attraction information management tasks specific for admin. This component supports admin to manage all attraction’s information within the park. The admin can add the attraction information as well as edit, delete the added of the attraction information and the existing of the attraction information. The display of attraction information is also included in this feature.

#### Collecting visiting statistics data for attraction information

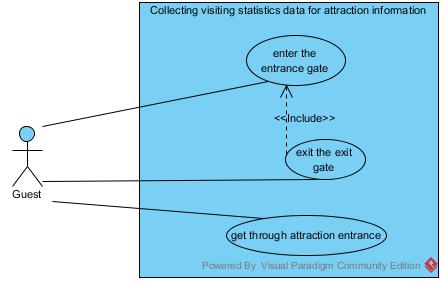


Figure-4 : Use case diagram to show collecting visiting statistics data for attraction information system overview

#### Attractions' statistic information

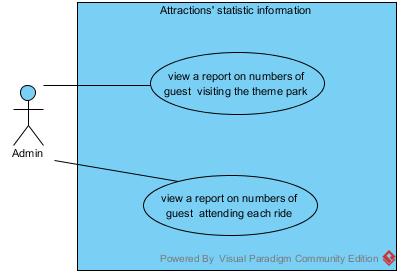


Figure-5 : Use case diagram to show attractions' statistic information system overview

## Use Case Description

#### User Case List of Guest Information Management System

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | Use case | | Primary Actor |
| UC-1 | | Register guest | Ticket Seller |
| UC-2 | | View guest information | Ticket Seller |
| UC-3 | | Edit guest information | Ticket Seller |
| UC-4 | | Delete guest information | Ticket Seller |

#### User Case List of Attraction Information Management System

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | Use case | | Primary Actor |
| UC-5 | | View attraction information from the system | Admin |
| UC-6 | | Add attraction information into the system | Admin |
| UC-7 | | Edit attraction information from the system | Admin |
| UC-8 | | Delete attraction information from the system | Admin |

#### User Case List of Collecting visiting statistics data for attraction information

| Use Case ID | Use case | | Primary Actor |
| --- | --- | --- | --- |
| UC-9 | | Enter the entrance gate by representing RFID tag at RFID reader | Guest |
| UC-10 | | Exit the exit gate by representing RFID tag at RFID reader | Guest |
| UC-11 | | Enter attraction entrance by representing RFID tag at RFID reader | Guest |

#### User Case List of Attractions' statistic information

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | Use case | | Primary Actor |
| UC-12 | | View a report on numbers of guest visiting the theme park within one week range | Admin |
| UC-13 | | View a report on numbers of guest attending each ride park within one week range | Admin |

### UC-1 Register guest into the system

**Use Case Description:**

|  |  |
| --- | --- |
| Use Case ID: | UC-1 |
| Use Case Name | Register guest |
| Short Description: | This function supports ticket seller and admin to register guest into the system. |
| Trigger: | Ticket seller choose 'Register guest' from the provided user interface |
| Related Use Case: | - |
| Actors: | Ticket Seller |
| Pre conditions: | - |
| Post conditions: | Guest information saved into the system. |
| Flow of Events: | **User Action:**  1. Ticket seller represent RFID tag in front of RFID reader  2. Ticket seller click on ‘Register new guest ’  3. Ticket seller goes to registration page.  4. Fill guest’s information into the form.  4.1. Fill guest’s name.  4.2. Fill guest’s email.  4.3. Fill guest’s gender  4.4. Fill guest’s age  5. Click on ‘Submit’ button  6. Display message 'Data inserted successfully' |
| Alternative and Exceptional: | From the normal flows number4.1. If user don’t fill in the name field  4.1.a. User will get the error message “Please fill out this field”.  From the normal flows number 4.1. If user input less than 5 or more than 30 letters.  4.1.b. User will get the error message “Name must be 5-30 characters in length”.  From the normal flows number 4.2. If user don’t fill in the mail field  4.2.a. User will get the error message “Please fill out this field”.  From the normal flows number 4.2. If user don’t fill in e-mail format.  4.2.b. User will get the error message “Please enter an email address”.  From the normal flows number 4.3. If user don’t fill in the phone number field  4.3.a. User will get the error message “Please fill out this field”.  From the normal flows number 4.3. If user don’t fill in the telephone number format.  4.3.a. User will get the error message “Invalid mobile no.”. |

**URS-1:** Ticket seller can register guest into the system.

**Introduction:** The admin and ticket seller can register guest into the system by scan the RFID tag and then fill the form.

**Input:**

|  |  |  |  |
| --- | --- | --- | --- |
| Input name | Description | Constraints | Example |
| Guest name | The name of the guest | A list of English letters, symbols and blanks.   * Must longer than 5 letters. * Not more than 30. * Can’t be Null. | Pipat saejuan |
| Guest e-mail | The e-mail of the guest | A list of English letters. Which is in the e-mail format. | pipat@camt.com |
| Guest phone number | The phone number of the guest | A list of 10 digits numbers. | 0812345678 |

**Action:**

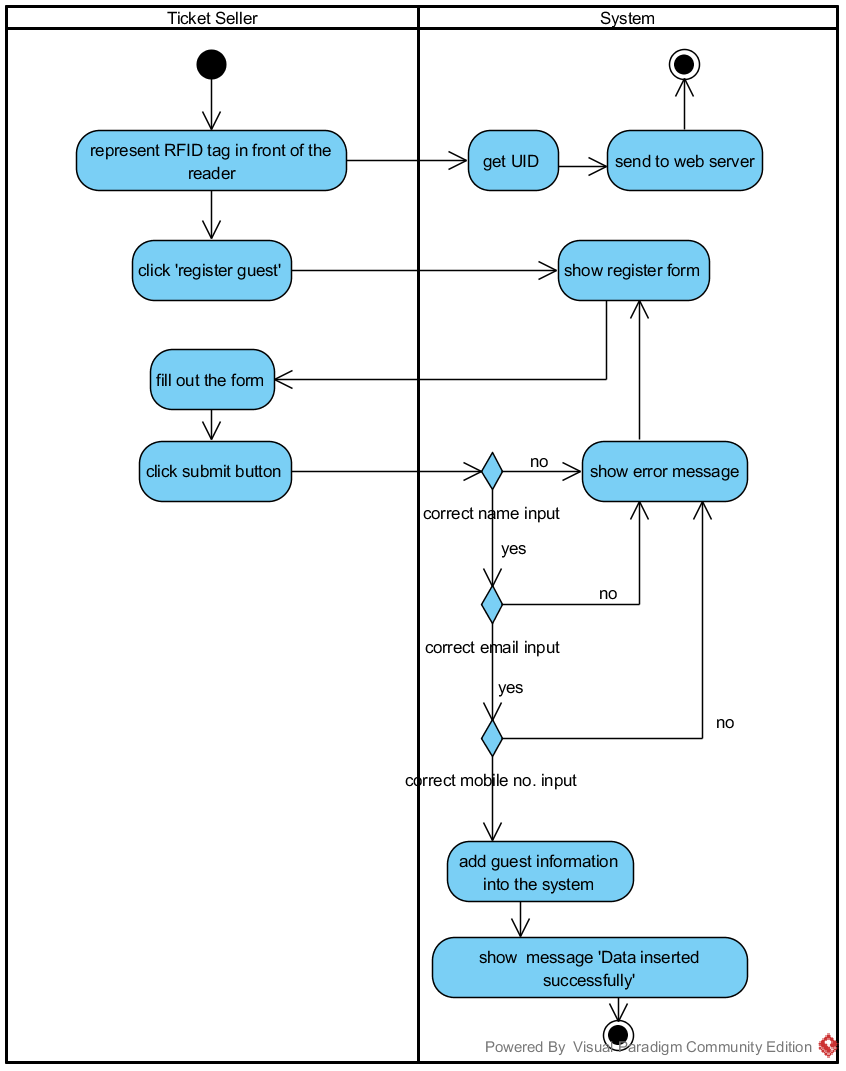
****

Figure-4 : AD-01, Activity diagram to show flow of URS-1

### UC-1 View guest information from the system

**Use Case Description:**

|  |  |
| --- | --- |
| Use Case ID: | UC-2 |
| Use Case Name | View guest information from the system |
| Short Description: | This function supports ticket seller to view the list of guest. |
| Trigger: | Ticket seller chooses 'View guest Info' from the provided user interface |
| Related Use Case: | <Edit guest information>, <Delete guest information> |
| Actors: | Ticket Seller |
| Pre conditions: | - |
| Post conditions: | Ticket Seller see guest information list. |
| Flow of Events: | **User Action:**  1. Click on ‘view guest information’  2. Show a list of guests on the screen |
| Alternative and Exceptional: | From the normal flows number 2 If the there are no attraction added yet.  3a. Empty table will be shown. |

**Action:**

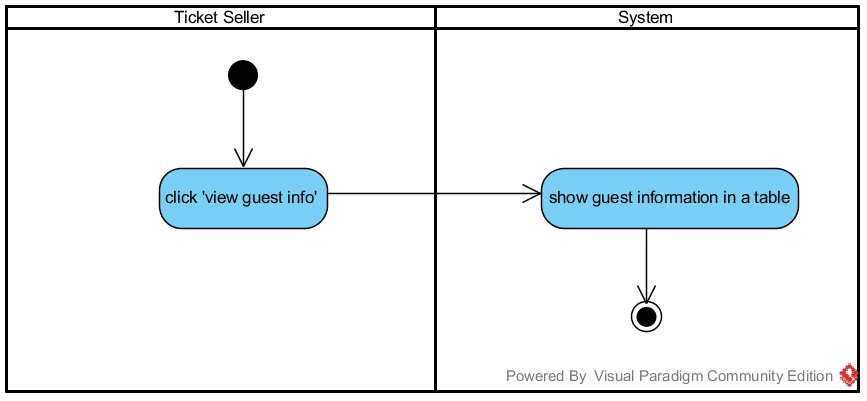
****

Figure-5: AD-02, Activity diagram to show flow of URS-2

### UC-3 Edit guest information from the system

**Use Case Description:**

|  |  |
| --- | --- |
| Use Case ID: | UC-3 |
| Use Case Name | Edit guest information from the system |
| Short Description: | This function supports ticket seller to edit guest information. |
| Trigger: | Ticket seller chooses 'View guest Info' from the provided user interface |
| Related Use Case: | <Edit guest information>, <View guest information> |
| Actors: | Ticket Seller |
| Pre conditions: | Ticket sellers have to register guest information into the database first. |
| Post conditions: | Updated information saves into the database. |
| Flow of Events: | **User Action:**  1. Click on ‘Edit’ button.  2. Show editing form.  3. Ticket sellers fulfills the information  4. Ticket sellers click on ‘update’ button  5. Save new update information into the system. |
| Alternative and Exceptional: |  |

**Input:**

| Input name | Description | Constraints | Example |
| --- | --- | --- | --- |
| Guest name | The name of the guest | A list of English letters symbols and blanks.   * Must longer than 5 letters. * Nomore than 15. * Can’t be Null. | chanthicha |
| Guest e-mail | The e-mail of the guest | A list of English letters. Which is in the e-mail format. | chanthicha@camt.com |
| Guest phone number | The phone number of the guest | A list of English letters. Which is in the telephone number format. | 081-2345678  053-890239 |

**Action:**

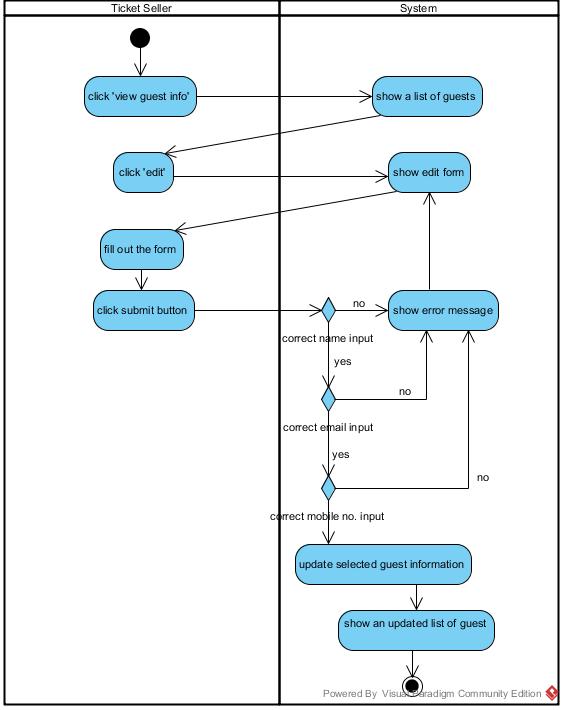


Figure-6: AD-03, Activity diagram to show flow of URS-3

**Output:** The updated information will save into the database.

### UC-4 Delete guest information from the system

**Use Case Description:**

|  |  |
| --- | --- |
| Use Case ID: | UC-4 |
| Use Case Name | Delete guest information from the system |
| Short Description: | This function support admin and ticket seller to delete guest from the system. |
| Trigger: | Admin/Ticket seller chooses 'View guest Info' from the provided user interface |
| Related Use Case: | <Register> |
| Actors: | Admin, Ticket Seller |
| Pre conditions: | Admin or ticket sellers have to register guest information into the database first. |
| Post conditions: | Information, which is chosen, gets deleting. |
| Flow of Events: | **User Action:**  1. Click on ‘delete’ button.  2. Show confirm dialog box.  3.Click 'OK'  4. Delete chosen guest information from the database. |
| Alternative and Exceptional: | From the normal flows number 3 If user click 'cancel' instead of 'OK'.  3a. Confirm dialog box is closed. |

**Action:**

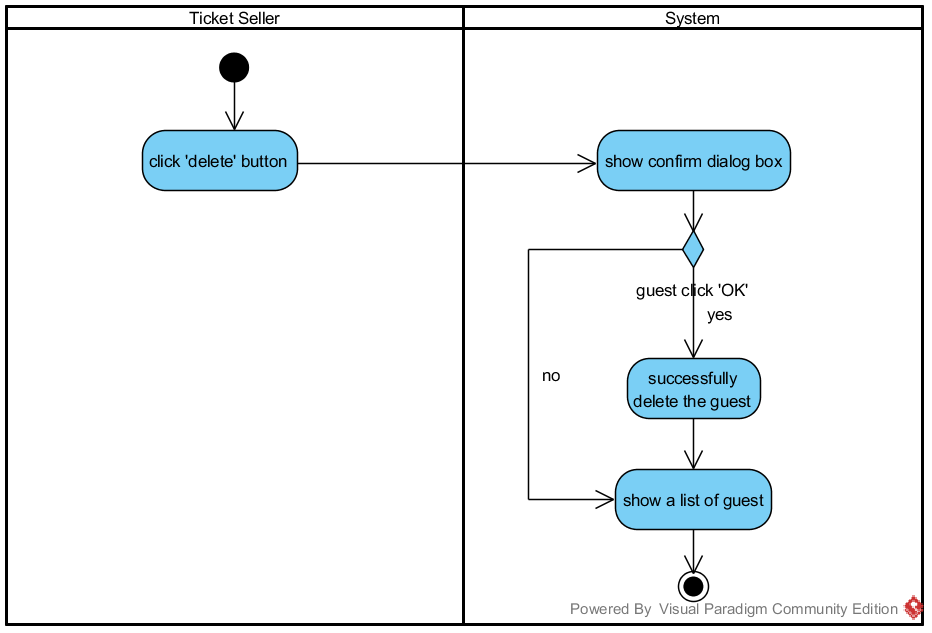


Figure-7: AD-04, Activity diagram to show flow of URS-4

### UC-5 View attraction information from the system

**Use Case Description:**

|  |  |
| --- | --- |
| Use Case ID: | UC-5 |
| Use Case Name | View attraction information from the system |
| Short Description: | This function supports admin to view the list of attraction. |
| Trigger: | Ticket seller choose 'View Attraction' from the provided user interface |
| Related Use Case: | <Edit attraction information>, <Delete attraction information>,  < View a report on numbers of guest attending each ride > |
| Actors: | Admin |
| Pre conditions: | - |
| Post conditions: | Admin see guest information list. |
| Flow of Events: | **User Action:**  1. Click on ‘view attraction information’  2. Show list of attraction on the screen |
| Alternative and Exceptional: | From the normal flows number 2 If the there are no attraction added yet.  3a. Empty table will be shown. |

**Action:**

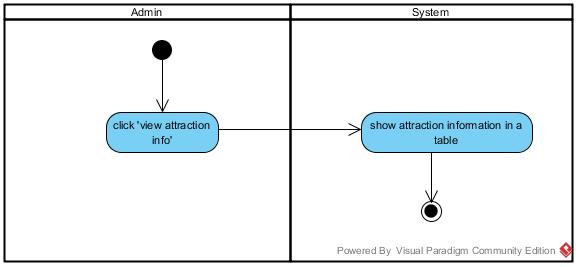
****

Figure-8: AD-05, Activity diagram to show flow of URS-5

### UC-6 Add attraction information into the system

|  |  |
| --- | --- |
| Use Case ID: | UC-6 |
| Use Case Name | Add attraction information into the system |
| Short Description: | This function supports admin to add attraction information into the system. |
| Trigger: | Admin choose 'Add Attraction' from the provided user interface |
| Related Use Case: | - |
| Actors: | Admin |
| Pre conditions: | - |
| Post conditions: | Attraction information saved into the system. |
| Flow of Events: | **User Action:**  1. Select 'add attraction'.  2. Show registration form on the screen.  3. Fill attraction’s information into the form.  3.1. Fill attraction’s name.  3.2. Fill attraction’s description.  3.3. Fill attraction’s height limit  4. Click on ‘Submit’ button  5. Save guest’s information into the database  6. Display message 'Data inserted successfully' |
| Alternative and Exceptional: | From the normal flows number 3.1. If user don’t fill in the name field  3.1.a. User will get the error message “The name field is required”.  From the normal flows number 3.1. If user input less then5 letters.  3.1.b. User will get the error message “The name field must not more than 5 characters”.  From the normal flows number 3.1. If user input more than 30 letters.  3.1.c. User will get the error message “The name field must less than 30 characters”.  From the normal flows number 3.2. If user don’t fill in the description field  3.2.a. User will get the error message “The description field is required”.  From the normal flows number 3.2. If user input less then5 letters.  3.2.b. User will get the error message “The description field must not more than 5 characters”.  From the normal flows number 3.2. If user input more than 300 letters.  3.2.c. User will get the error message “The description field must less than 300 characters”.  From the normal flows number 3.3. If user don’t fill in the height limit field  3.3.a. User will get the error message “The height limit field is required”.  From the normal flows number 3.3. If user not fill the height limit field in numeric format.  3.3.a. User will get the error message “The height limit must be number only”. |

**Input:**

| Input name | Description | Constraints |  | Example |
| --- | --- | --- | --- | --- |
| Attraction name | The name of each attraction | A list of English letters symbols and blanks.   * Must longer than 5 letters. * Not more than 30. * Can’t be Null. |  | MontuMontu |
| Attraction description | The description of each attraction | A list of English letters symbols and blanks.   * Must longer than 5 letters. * Not more than 300. * Can’t be Null. |  | This attraction is a roller coaster, which is very x-theme. |
| Attraction height limit | The height limit of each attraction | A list of numeric.   * Can’t be Null. |  | 154 |

**Action:**

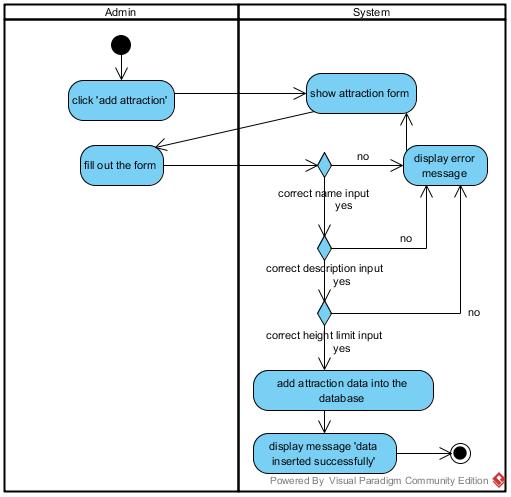
****

Figure-9: AD-06,Activity diagram to show flow of URS-6

### UC-7 Edit attraction information from the system

|  |  |
| --- | --- |
| Use Case ID: | UC-7 |
| Use Case Name | Edit attraction information from the system |
| Short Description: | This function supports admin to edit attraction information. |
| Trigger: | Admin chooses 'View Attraction' from the provided user interface |
| Related Use Case: | <Add attraction> |
| Actors: | Admin |
| Pre conditions: | Admin have to add attraction information into the database first. |
| Post conditions: | Updated information saves into the database. |
| Flow of Events: | **User Action:**  1. Click on ‘Edit’ button.  2. Show editing form.  3. Admin fulfills the information  3.1. Fill attraction’s name.  3.2. Fill attraction’s description.  3.3. Fill attraction’s height limit  4. Admin click on ‘update’ button  5. Save new update information into the system. |
| Alternative and Exceptional: | From the normal flows number 3.1. If user don’t fill in the name field  3.1.a. User will get the error message “Please fill out this field”.  From the normal flows number 3.1. If user input less then5 letters.  3.1.b. User will get the error message “Name must be 5-30 characters in length”.  From the normal flows number 3.1. If user input more than 30 letters.  3.1.c. User will get the error message “Name must be 5-30 characters in length s”.  From the normal flows number 3.2. If user don’t fill in the description field  3.2.a. User will get the error message “Please fill out this field”.  From the normal flows number 3.2. If user input less then5 letters.  3.2.b. User will get the error message “Description must be 5-60 characters in length”.  From the normal flows number 3.2. If user input more than 300 letters.  3.2.c. User will get the error message “Description must be 5-60 characters in length”.  From the normal flows number 3.3. If user don’t fill in the height limit field  3.3.a. User will get the error message “Please fill out this field”.  From the normal flows number 3.3. If user not fill the height limit field in numeric format.  3.3.a. User will get the error message “Please enter a number”. |

**Input:**

|  |  |  |
| --- | --- | --- |
| Input name | Description | Example |
| Attraction name | The name of attraction | MontuMontu |
| Attraction description | The description of attraction | This attraction is a roller coaster, which is very x-theme. |
| Attraction height | The height of attraction | 154 |

**Action:**

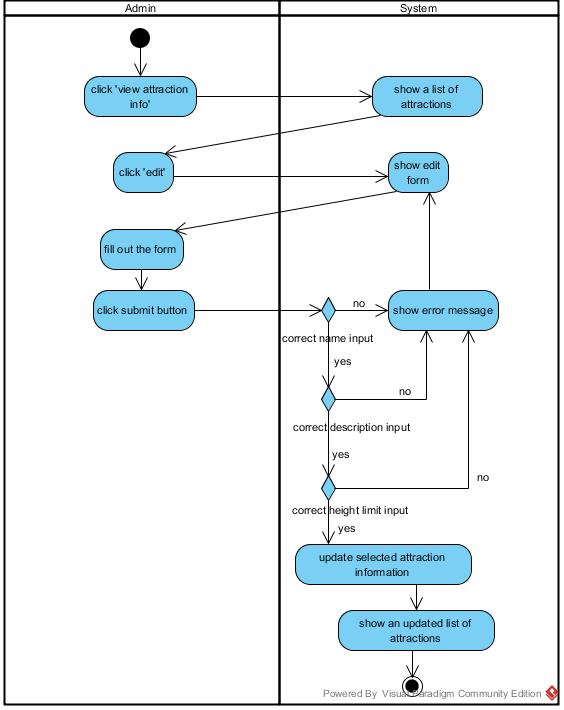


Figure-10: AD-07, Activity diagram to show flow of URS-7

### UC-8 Delete attraction information from the system

|  |  |
| --- | --- |
| Use Case ID: | UC-8 |
| Use Case Name | Delete attraction information from the system |
| Short Description: | This function support admin to delete guest from the system. |
| Trigger: | Admin chooses 'View Attraction' from the provided user interface |
| Related Use Case: | <Add attraction>  <View attraction> |
| Actors: | Admin |
| Pre conditions: | 1. Admin have to add attraction information into the database first.  2. Admin must choose to view attraction information first. |
| Post conditions: | Information, which is chosen, will be deleted. |
| Flow of Events: | **User Action:**  1. Click on ‘delete’ button.  2. Show confirm dialog box.  3.Click 'OK'  4. Delete chosen guest information from the database. |
| Alternative and Exceptional: | From the normal flows number 3 If user click 'cancel' instead of 'OK'.  3a. Confirm dialog box is closed. |

**Action:**

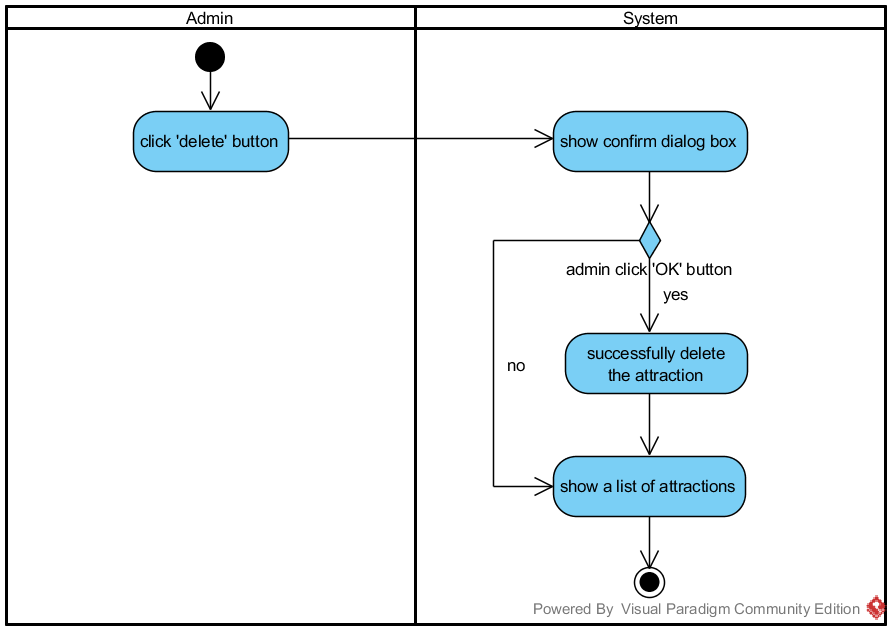


Figure-15: AD-08, Activity diagram to show flow of URS-8

### UC-9 Enter the entrance gate by representing RFID tag at RFID reader

|  |  |
| --- | --- |
| Use Case ID: | UC-9 |
| Use Case Name | Enter the entrance gate by presenting RFID tag at RFID reader |
| Short Description: | This function support the system to collect The UID from the guest in order to create statistic data of number of guest visiting the theme park. |
| Trigger: | Guest presents the RFID tag at the reader |
| Related Use Case: | none |
| Actors: | Guest |
| Pre conditions: | none |
| Post conditions: | The UID of guest, date and time will be stored in a table in the database. |
| Flow of Events: | **User Action:**  1. Guest represents The RFID tag in front of a reader at the entrance gate.  2. System shall read the UID from the guest's RFID tag.  3. System shall send the UID to the web application.  4. System shall store the UID into the database. |
| Alternative and Exceptional: | none |

**Action:**

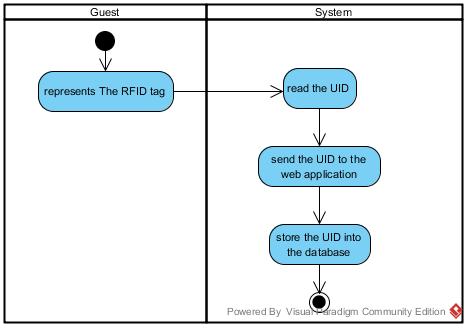


Figure-16: AD-09, Activity diagram to show flow of URS-09

### UC-10 Exit the exit gate by representing RFID tag at RFID reader

|  |  |
| --- | --- |
| Use Case ID: | UC-10 |
| Use Case Name | Exit the exit gate by representing RFID tag at RFID reader |
| Short Description: | This function support the system to collect The UID from the guest in order to create a list of information for guest viewing. |
| Trigger: | Guest presents the RFID tag at the reader |
| Related Use Case: | none |
| Actors: | Guest |
| Pre conditions: | none |
| Post conditions: | The UID of guest, date and time will be stored in a table in the database. |
| Flow of Events: | **User Action:**  1. Guest represents The RFID tag in front of a reader at the exit gate.  2. System shall read the UID from the guest's RFID tag.  3. System shall send the UID to the web application.  4. System shall store the UID into the database. |
| Alternative and Exceptional: | none |

**Action:**

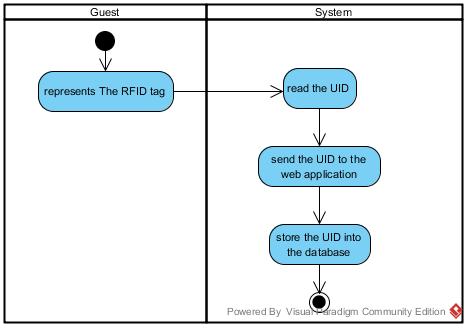


Figure-17: AD-09, Activity diagram to show flow of URS-10

### UC-11 Enter attraction entrance by representing RFID tag at RFID reader

|  |  |
| --- | --- |
| Use Case ID: | UC-11 |
| Use Case Name | Enter attraction entrance by representing RFID tag at RFID reader |
| Short Description: | This function support the system to collect The UID from the guest in order to create statistic data of number of guest visiting the theme park and a list of information for guest viewing. |
| Trigger: | Guest presents the RFID tag at the reader |
| Related Use Case: | <View guest's past activity> |
| Actors: | Guest |
| Pre conditions: | none |
| Post conditions: | The UID of guest, date and time will be stored in a table in the database. |
| Flow of Events: | **User Action:**  1. Guest represents The RFID tag in front of a reader at the entrance of the attraction.  2. System shall read the UID from the guest's RFID tag.  3. System shall send the UID to the web application.  4. System shall store the UID into the database.4. System shall store the UID into the database. |
| Alternative and Exceptional: | none |

**Action:**

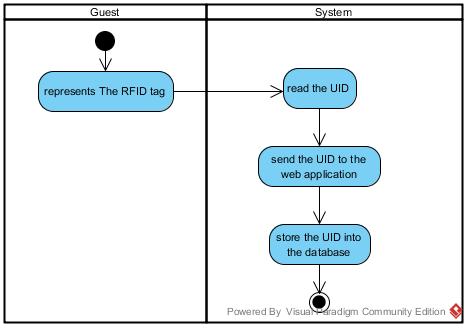


Figure-18: AD-09, Activity diagram to show flow of URS-11

### UC-12 View a report on numbers of guest visiting the theme park within one week range.

|  |  |
| --- | --- |
| Use Case ID: | UC-12 |
| Use Case Name | View a report on numbers of guest visiting the theme park within one week range. |
| Short Description: | This function support the admin to view the result number of guest visiting the theme park in a span of one week. |
| Trigger: | Admin chooses 'Activity' from the provided user interface |
| Related Use Case: | <Enter entrance gate by representing RFID tag> |
| Actors: | Admin |
| Pre conditions: | none |
| Post conditions: | none |
| Flow of Events: | **User Action:**  1. Admin select 'Activity' from the menu.  2. System shall retrieve the date and sym of total visitors from the database.  3. System shall provide a line chart to display the number of visitors in current and past 6 days. |
| Alternative and Exceptional: | From the normal flows number 2. If the data table is empty  2.a The system shall display the message "The activity record is empty". |

**Action:**

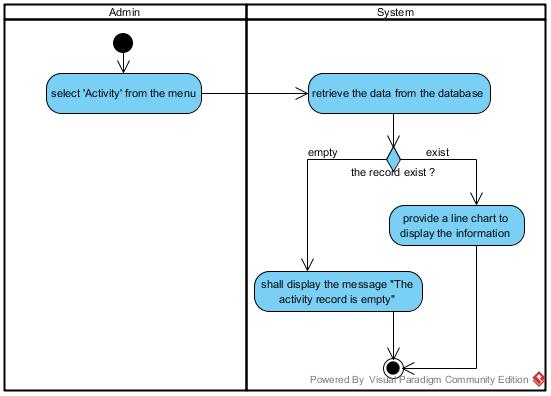


Figure-19: AD-10, Activity diagram to show flow of URS-12

### UC-13 View a report on numbers of guest attending each ride within one week range

|  |  |
| --- | --- |
| Use Case ID: | UC-12 |
| Use Case Name | View a report on numbers of guest attending each ride within one week range |
| Short Description: | This function support the admin to view the result numbers of guest attending rides in the theme park within a span of one week. |
| Trigger: | Admin chooses 'View Attraction' from the provided user interface |
| Related Use Case: | <Enter attraction entrance by representing RFID tag>  <View attractions information> |
| Actors: | Admin |
| Pre conditions: | none |
| Post conditions: | none |
| Flow of Events: | **User Action:**  1. Admin select 'View Attraction' from the menu.  2. System shall provide the user interface for users to select an attraction.  3. Admin select an attraction.  4. System shall retrieve the date and sym of total visitors from the database.  5. System shall provide a line chart to display the number of visitors in current and past 6 days. |
| Alternative and Exceptional: | From the normal flows number 4. If the data table is empty  4.a The system shall display the message "The activity record is empty". |

**Action:**

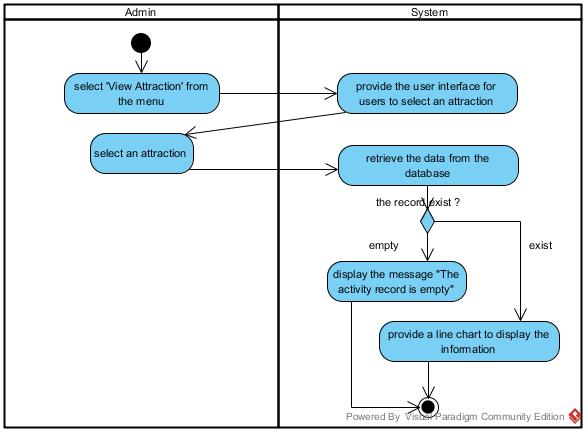


Figure-20: AD-11, Activity diagram to show flow of URS-13