

GURU NANAK INSTITUTE OF TECHNOLOGY



INFORMATION TECHNOLOGY

NAME - TEERATH KUNDU

ROLL NO. - 500421010092

SUBJECT - SOFTWARE ENGINEERING LAB

CODE - IT 494

YEAR - 2ND YEAR

SEMESTER - 4TH SEMESTER

Table of Contents:

| SL. NO | Date of Experiment | Name of Experiment | Remarks/Sign |
|--------|--------------------|--|--------------|
| 1 | 29-04-2023 | Introduction, Overall description, Specific Requirements | |
| 2 | 30-04-2023 | Draw a level 0 DFD and level 1 DFD for Gas booking system. Define it also. | |
| 3 | 01-05-2023 | Draw a use-case diagram and class diagram for Gas booking system. Define it also. | |
| 4 | 02-05-2023 | Draw an activity diagram and component diagram for Gas booking system. Define it also. | |
| 5 | 03-05-2023 | Draw a deployment diagram and sequence diagram for Gas booking system. Define it also. | |
| 6 | 04-05-2023 | COCOMO exercise | |

EXPERIMENT : 01

Introduction

➤ Purpose:

This SRS is about the website for gas booking system. This website is an online gas booking website through which a user can access its information and manage all the adding, updating, deleting the assets and some of its tasks. The admin user can change update the information regarding gas selling and buying and cancellation.

➤ Scope:

It can be used in any LPG distribution center communicating with consumer via internet.

➤ Definitions, acronyms, and abbreviations:

NA – Not Applicable

GBS – Gas Booking System

SRS – Software Requirement Specification

➤ References:

The copy of GBS manual

➤ Overview:

This SRS is organized as follows:

1. Gives an overall idea of product
2. Describes the specific requirement of the GBS

Overall description

➤ Product perspective:

- **Admin's interface:** Admin is a person who will handle the entire website. For that person has to give the user's name and password to enter the admin page. After entering right password admin person can enter the admin home page.
- **User's interface:** User can visit the home page of real estate in which first the introduction of the site is mentioned first. The registered user can login from the login module and new user can register from the registered module. User can select the gas and book a gas cylinder and pay the bill.

➤ Product functions:

→ Login profile

→ Gas cylinder booking

→ Payment

→ Delivery

➤ User characteristics:

Normal user – No qualification required to access it. Anyone who is able may use website.

Designers – Users who may take interest in the design of the website and its functionality.

Developers – Users who may try to understand the functionality of the website.

Administrator – Users with highest privileges who can modify anything on the website.

Intermediate users – These users are those who connect the normal users to the booking/other functionalities.

➤ Constraints:

- **Recommended RAM 1GB (min):** The CSS user in the website is dependent on the hardware. If low RAM then there might be performance issues.
- **JavaScript Enabled in browsers:** Make sure JavaScript is enabled in the browsers as some UI may need it.

➤ **Assumptions and dependencies:**

Assumptions:

- The code should be free with compilation errors/syntax errors.
- The product must have an interface which is simple enough to understand.

Dependencies:

- All necessary hardware and software are available for implementing and use of the tool.
- The proposed system would be designed, developed and implemented based on the software requirements specifications document.
- End users should have basic knowledge of computer and we also assure that the users will be given software training documentation and reference material.

Specific Requirements

➤ **External interfaces requirements:**

Hardware Requirement:

- PentiumIV 400MHz and Above
- Minimum 1GB RAM
- 15" Color Monitor
- Keyboard
- Mouse

Software Requirement:

- Browser – Chrome/Firefox/Safari/IE 8 or higher
- Operating System – Windows XP/7/Vista/8/10
- Web Server – Apache
- DBMS – MySQL Server
- Internet Tools – HTML, JavaScript, JQuery, Ajax

➤ **Functions requirements:**

Since this project uses database and control, it needs the retrieval of information from the database. It needs access of database from a front end as ASP.net is a Microsoft Family product, it provides easy linking to the database, along with the flexibility to develop a user-friendly front end.

- **Usability:** The interface should use term and concepts, which are drawn from the experience of the people who will make most of the system.
- **Efficiency:** The system must provide easy and fast access without consuming more cost.
- **Reliability:** User should never be surprised by the behaviour of the system and it's easy to use to stored data and easy to used transfer voice data.

➤ **Performance requirements:**

The performance is best observed above 40 kbps. It is recommended that if the website is to be operated, then minimum of 40 kbps speed is needed.

Other requirements are just as java plug ins must be available, CSS must work properly and Google chrome browser is best suited.

➤ **Design constraints:**

- GUI is only in English
- Login and password are used for identification of user and there is no facility for guest

➤ **Software system attributes:**

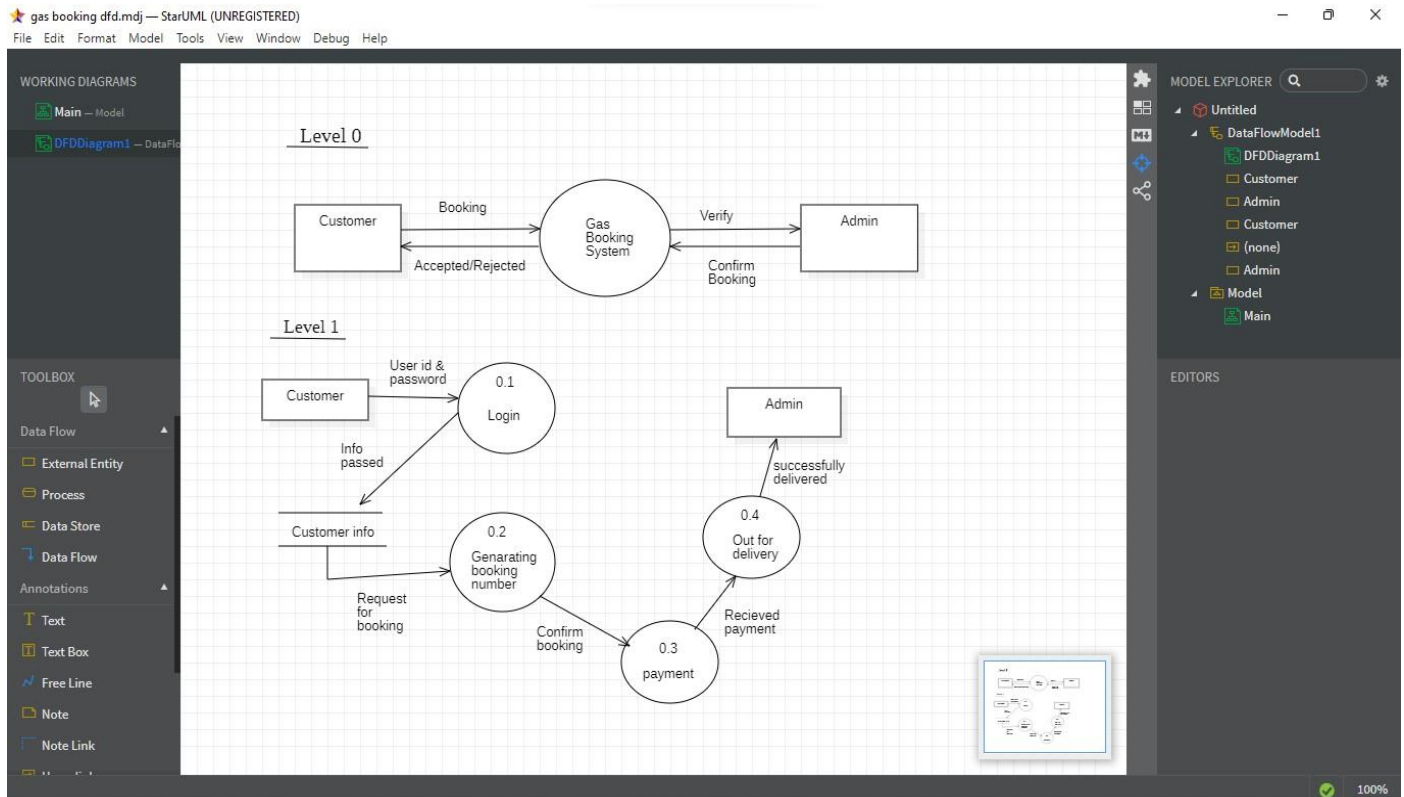
- **Reliability:** This system is designed to have simple database just to cater the exact need of real estate management. It is tested for all the constraints at development stage.
- **Availability:** This system will only be available till the system on which it is installed is running.
- **Security:** The system is provided with authentication without which no user can pass. So only the legitimate users are allowed to use the application. If the legitimate users share the authentication information the system is open to outsiders.
- **Maintainability:** There will be no maintenance required for the software. The database is provided by the end-user and therefore is maintained by this user.
- **Portability:** The system is not portable as it is a standalone application running on single system with no shared database.

➤ **Other requirements:**

- **Database** – A database is must to store all the details of the users and put some validations on them.
- **Server** – It is assumed that a server is processing all the requests from the site and giving out the desired output to the end user.

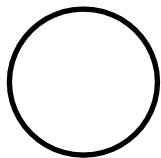
EXPERIMENT : 02

Draw a level 0 DFD and level 1 DFD for Gas booking system. Define it also.



A DFD is a hierarchical graphical model of a system that shows the different processing activities or functions that the system performs and the data interchange among those functions.

There are essentially different types of symbols used for constructing DFDs. They are as follows:



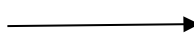
→ Process



→ External entity



→ Data store



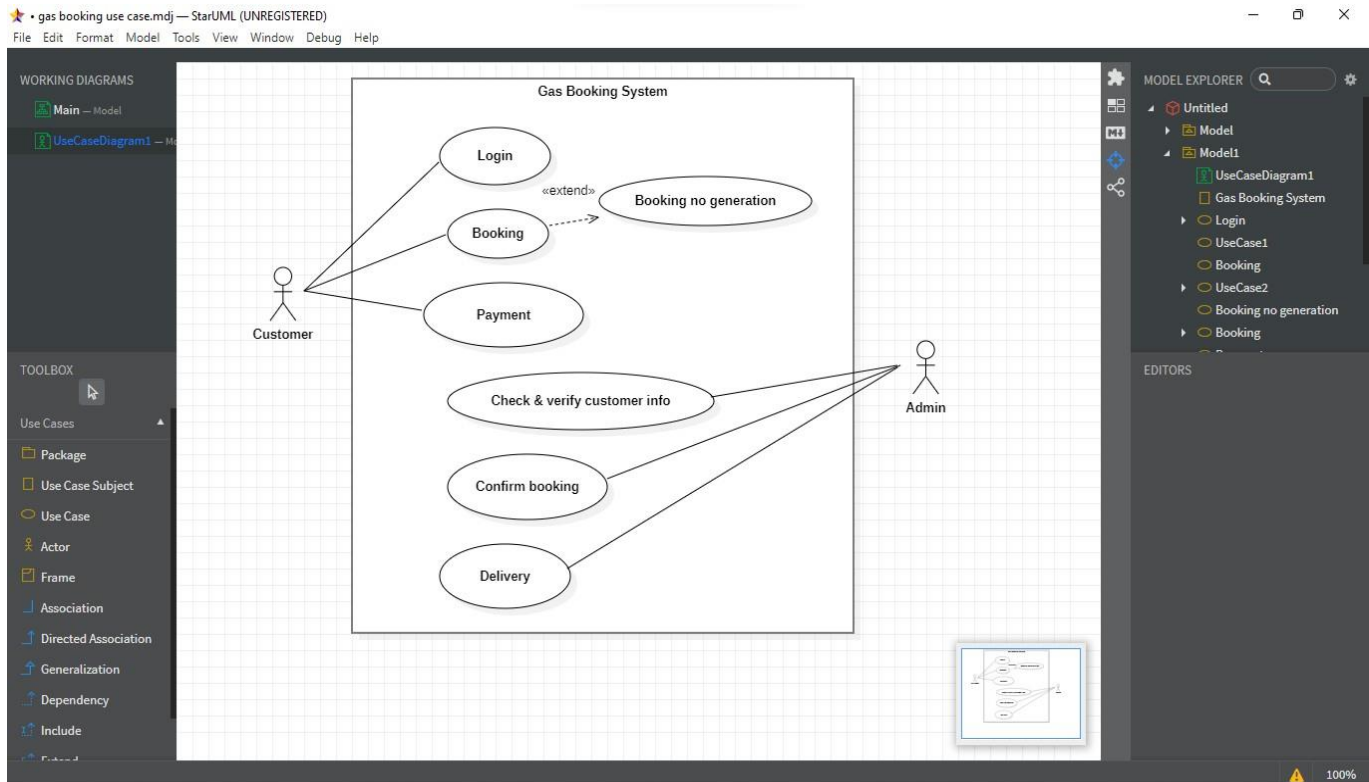
→ Data flow

In the above DFD a gas office needs to develop a software that would help it to book gas cylinder through online. In context diagram (level 0) there is one process and two external entities. In level 1 diagram a customer who have already registered would have first login by his/her valid user id and password. Each registered customer can book gas cylinder and then a booking number will be generated. Gas cylinder will be delivered after clearance of payment.

EXPERIMENT : 03

Draw a use-case diagram and class diagram for Gas booking system. Define it also.

USE-CASE DIAGRAM



Intuitively, the use cases represent the different ways in which a system can be used by the users.
Text description for above use case diagram:

Case 1 –

Login

Step 1: Customer selects login option.

Step 2: If already registered enter valid user id and password else customer have to register first.

Step 3: System displays the customer details.

Case 2 –

Booking

Step 1: Customer selects gas book option.

Step 2: System generates booking number.

Case 3 –

Payment

Step 1: customer selects payment option and complete payment process.

Case 4 –

Check & verify customer info

Step 1: System passes all data's.

Step 2: Admin check and verify this.

Case 5 –

Confirm booking

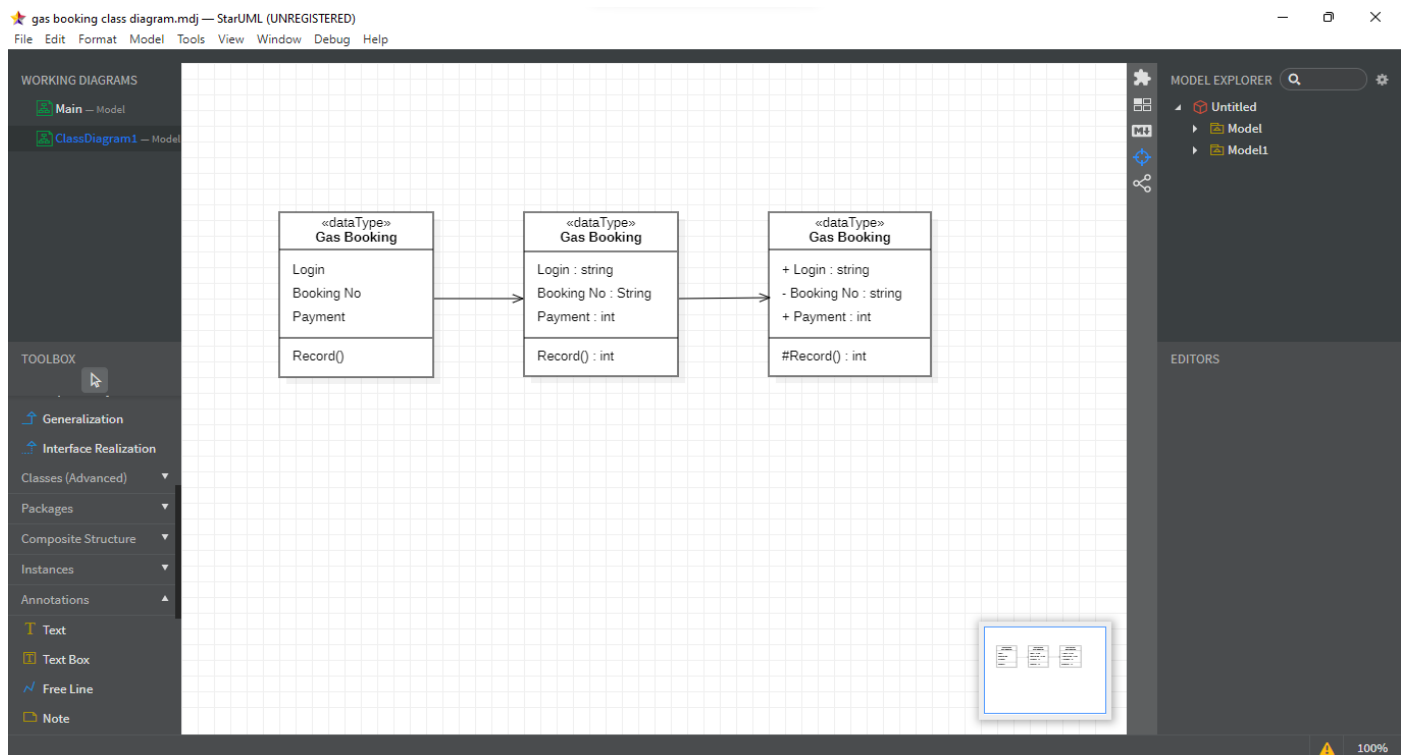
Step 1: Admin confirms the booking.

Case 6 –

Delivery

Step 1: Admin delivers the gas cylinder.

CLASS DIAGRAM



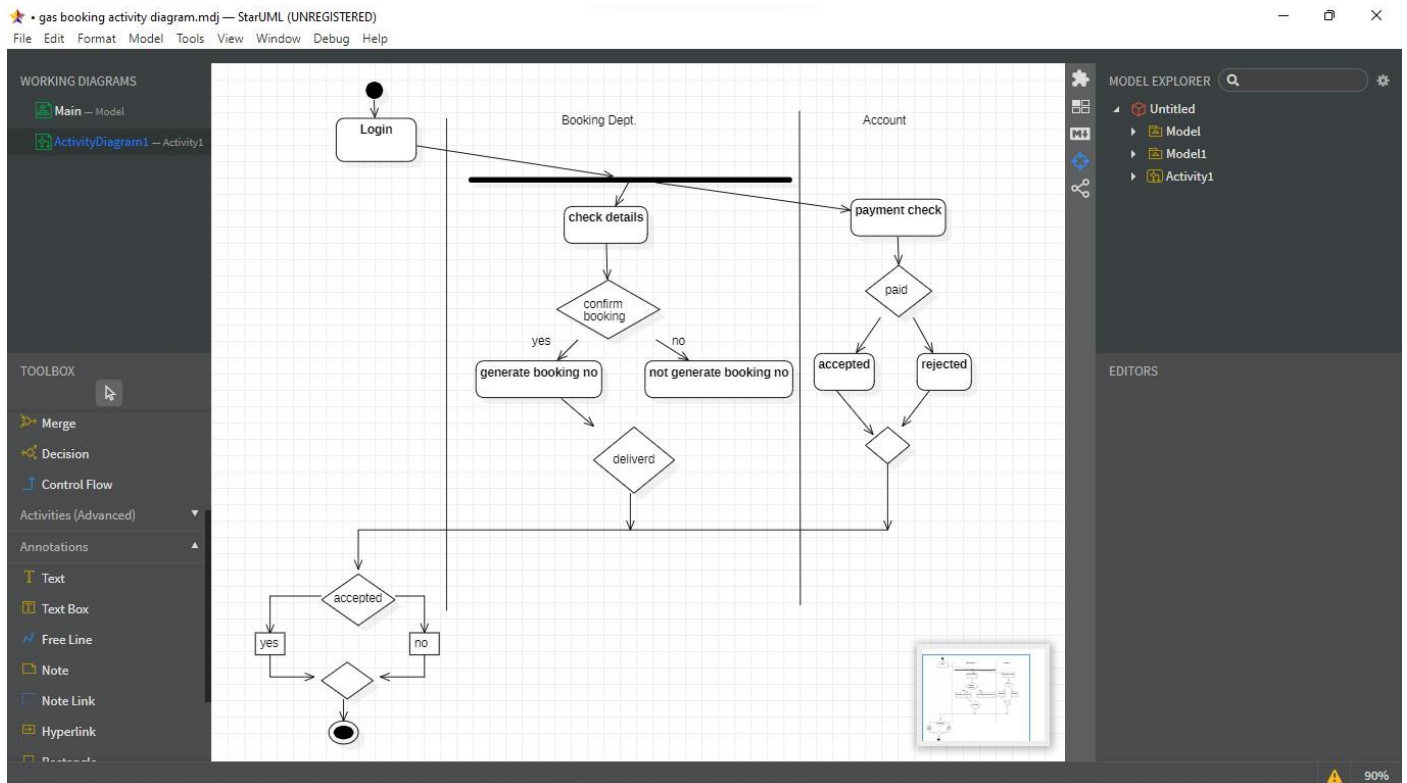
This is the most used UML diagram in the field of software engineering design. It is called as a main building block of any object-oriented solution. Usually it illustrates is a system, attributes and operations of each class and also the relationship between each class.

In the above Class Diagram classes are represented as solid outline rectangles with compartments. 1st compartment contains name of the class that is “Gas Booking”. 2nd compartment contains objects of class named “Login”, “Booking no”, and “Payment”. 3rd compartment contains methods.

EXPERIMENT : 04

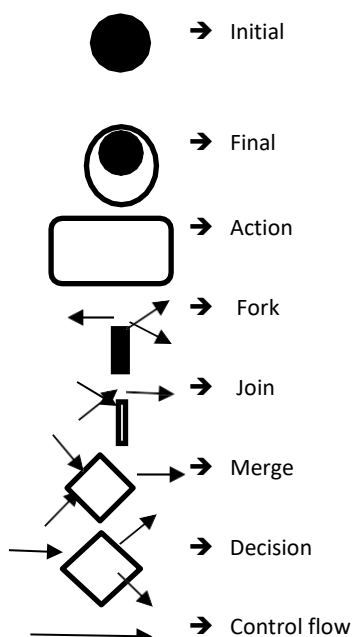
Draw an activity diagram and component diagram for Gas booking system. Define it also.

ACTIVITY DIAGRAM



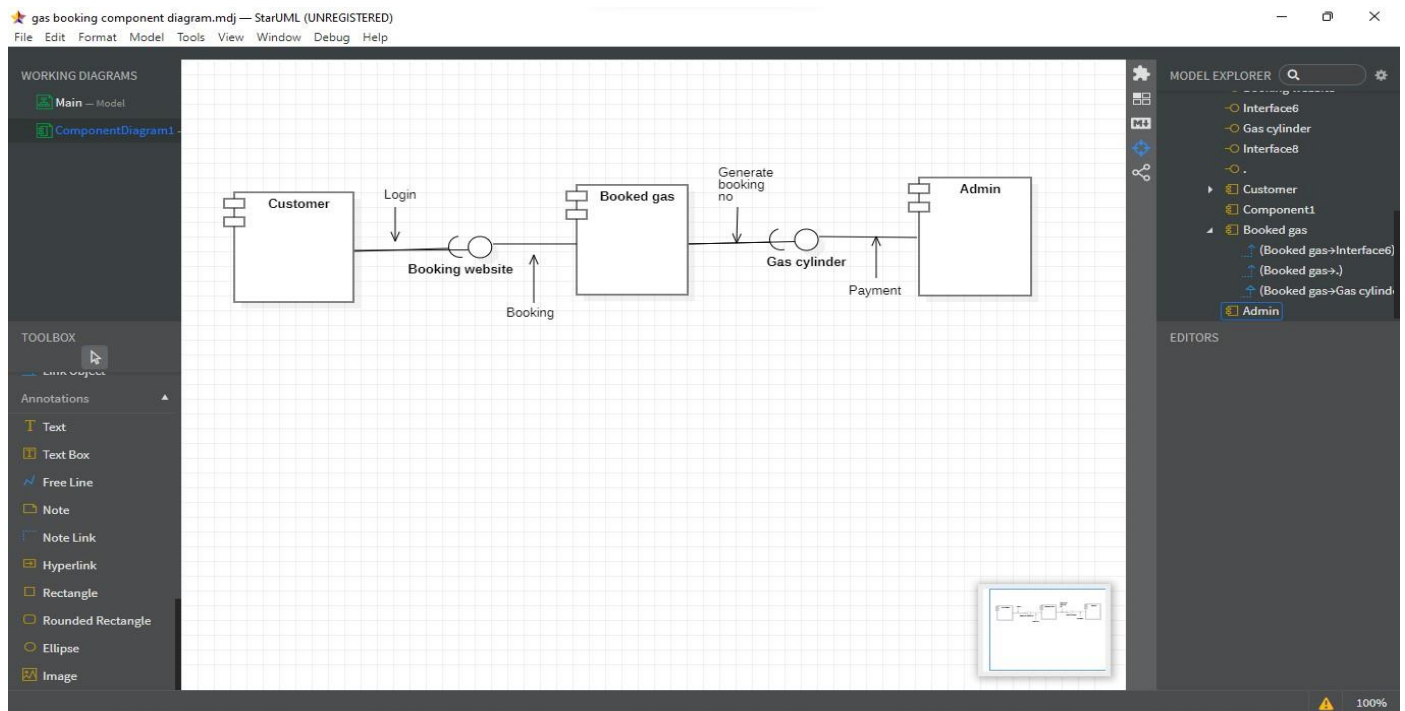
Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity diagram focuses on representing various activities or chunks of processing and their sequence of activation.

There are essentially different types of symbols used for constructing Activity Diagram. They are as follows



The online gas booking process is shown as an activity diagram in the above figure. There are two segments in this Activity Diagram, one is Booking Dept where the booking process is shown and another is Account where the payment process is shown.

COMPONENT DIAGRAM



A component diagram can be used to represent the physical structure of an implementation in terms of the various components of the system. A component diagram is typically used to achieve the following purposes:

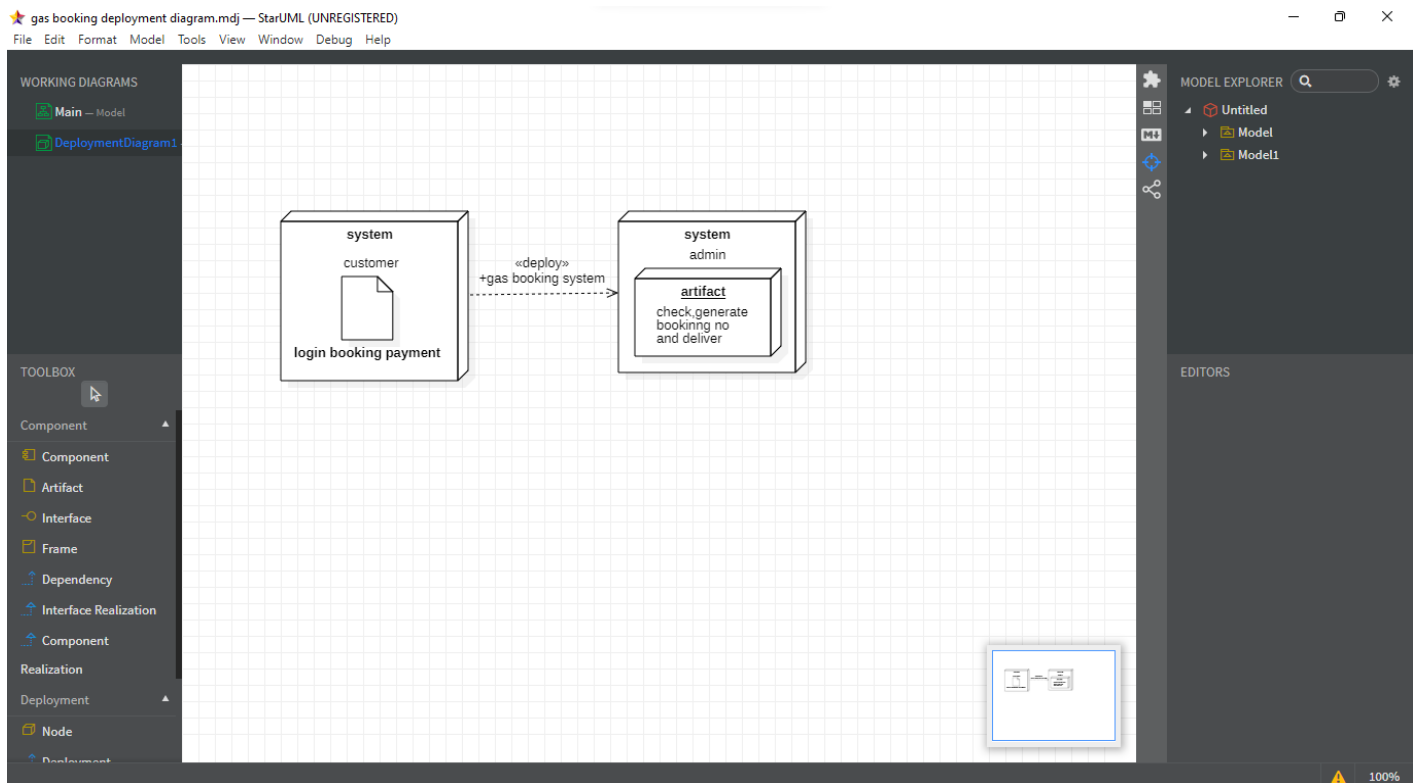
- Organize source code to be able to construct executable releases.
- Specify dependencies among different components.

The online gas booking process is shown as a Component Diagram in the above Figure. There are 3 components Customer, Booked Gas and Admin in this Component Diagram.

EXPERIMENT : 05

Draw a deployment diagram and sequence diagram for Gas booking system. Define it also.

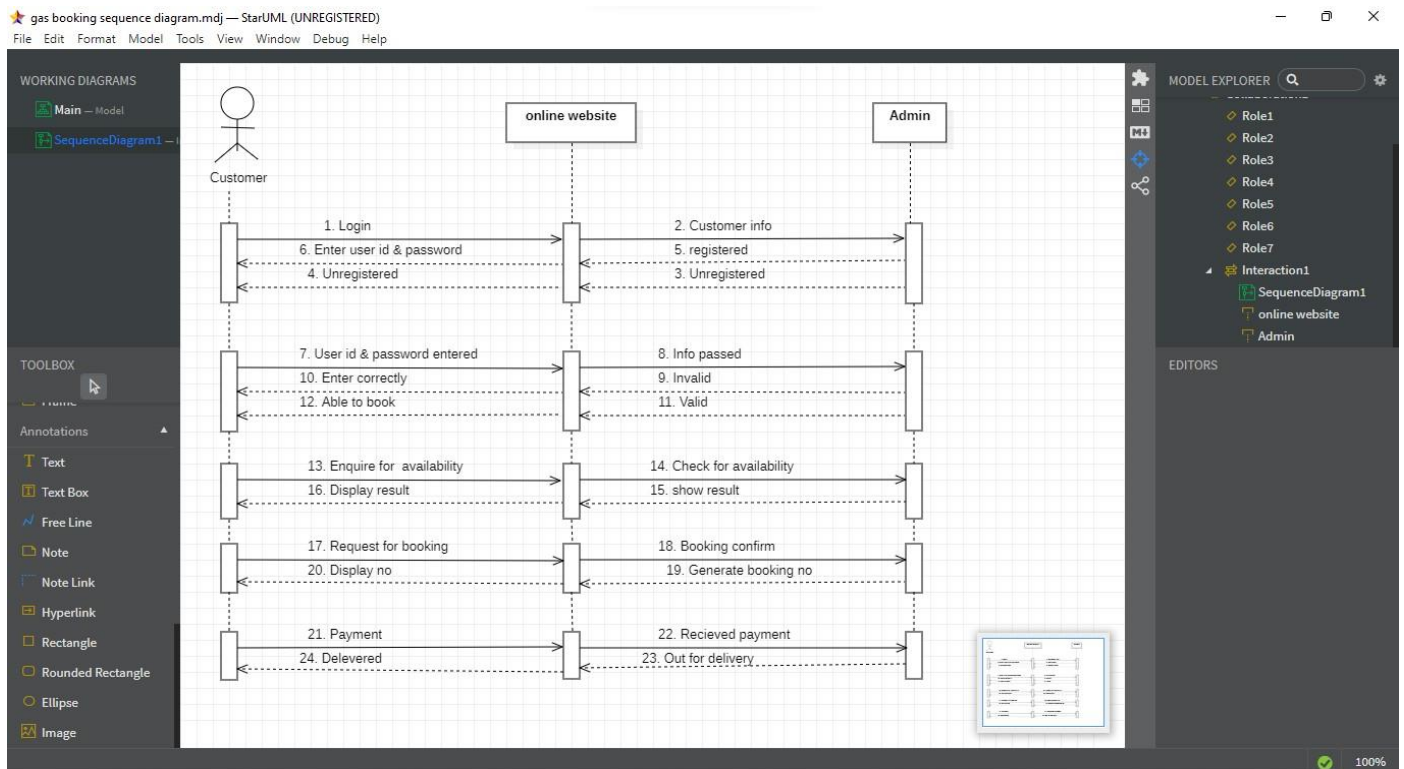
DEPLOYMENT DIAGRAM



The deployment diagram shows the environmental view of a system. That is, it captures the environment in which the software solution is implemented. In other words, a deployment diagram shows how a software system will be physically deployed in the hardware environment. That is, which component will execute on which hardware component and how they will they communicate with each other.

The online gas booking process is shown as a Deployment Diagram in the above Figure.

SEQUENCE DIAGRAM



A sequence diagram shows interaction among objects as a two-dimensional chart. The chart is read from top to bottom. The objects participating in the interaction are shown at the top of the chart as boxes attached to a vertical dashed line. Inside the box the name of the object is written.

The online gas booking process is shown as a Sequence Diagram in the above Figure. 3 objects Customer, Online Website and Admin are shown in the diagram. We observe that the exact objects which participate to realise the gas booking behaviour and the order in which they interact can be clearly inferred from the sequence diagram.

EXPERIMENT : 06

COCOMO exercise ->

Problem: Considering your immense expertise in software development, The Absolute Beginners Inc. has recently Allotted you amega project. The goal of the project is to create a database of all Hindi films released since 2000. The software would allow one to generate a list of top ten hit films, top ten flop films, best comedy Films, and so on. Using your prior experience, you have decided the approximate sizes of each module of the Software as follows:

- Data entry (0.9 KDSI)
- Data update (0.7 KDSI)
- Query (0.9 KDSI)
- Report generation and display (2 KDSI)
- Also take into consideration the following cost drivers with their ratings:
- Storage constraints (Low)
- Experience in developing similar software (High)
- Programming capabilities of the developers (High)
- Application of software engineering methods (High)
- Use of software tools (High) (All other cost drivers have nominal rating).

Now answer the following:

1. Applying intermediate COCOMO estimate the minimum size of the team you would require to develop this System
2. Assuming that your client would pay Rs. 50,000 per month of development, how much would be the likely Billing?

Solution: -

The COCOMO estimation formula for organic software is

$$\text{Effort} = a * (\text{KDSI})^b \text{ PM}$$

$$\text{Tdev} = a * (\text{Effort/corrected})^c \text{ Months}$$

We know,

$$a = 2.4$$

$$b = 1.05$$

$$c = 0.38$$

Adding all KDSI we get,

$$\text{KDSI} = 0.9 + 0.7 + 0.9 + 2 = 4.5$$

Therefore,

$$\text{Effort} = 2.4 * (4.5)^{1.05} \text{ PM} = 11.64 = 12 \text{ PM (approx.)}$$

Total weight for the cost driver

$$\text{EAF} = 1.00 * 0.91 * 0.95 * 0.91 * 0.91 = 0.7158 = 0.72 \text{ (approx.)}$$

$$\text{Effort/corrected} = \text{Effort} * \text{EAF} = 12 * 0.72 = 8.64 = 9 \text{ PM (approx.)}$$

$$\text{Tdev/corrected} = 2.4 * 9^{0.38} \text{ Month} = 5.531 = 6 \text{ Month (approx.)}$$

Therefore,

The minimum size of development team

$$= \text{Effort/Tdev} = 9/6 = 1.5 = 2 \text{ Person (approx.)}$$

$$\text{Total amount} = 6 * 50,000 = 3,00,000$$