

QR Code Generation And Consumption Document

Contents

QR Code :	3
Requirements for generating QR Code :	4
Steps To Generate QR Code:-	5
Requirements for generating QR Code :	6
References to be used:	6
Web Config Values :	6
Taking Backup of Original File :	7
Generating QR Code :	8
Finding Location to stamp PDF with QR Code:	9
Stamping PDF with QR Code:	10
QR Code Samples :	11
List of Original PDF and PDF with QR Code :	11
Original PDF :	12
PDF with QR Code :	13

QR Code :



QR code (abbreviated from Quick Response Code) is the trademark for a type of matrix barcode (or two-dimensional barcode) first designed in 1994 for the automotive industry in Japan. A barcode is a machine-readable optical label that contains information about the item to which it is attached. A QR code uses four standardized encoding modes (numeric, alphanumeric, byte/binary, and kanji) to efficiently store data; extensions may also be used.

The Quick Response (QR code) system became popular outside the automotive industry due to its fast readability and greater storage capacity compared to standard UPC barcodes. Applications include product tracking, item identification, time tracking, document management, and general marketing.

A QR code consists of black squares arranged in a square grid on a white background, which can be read by an imaging device such as a camera, and processed using Reed–Solomon error correction until the image can be appropriately interpreted. The required data is then extracted from patterns that are present in both horizontal and vertical components of the image.

Requirements for generating QR Code :

- itextsharp.dll
- QRCoder.dll
- Spire.License.dll
- Spire.Pdf.dll

Three third party tools are used for generating QR Code and there uses are as follows :-

1. itextsharp.dll

To read pdf file and to stamp the generated QR code on the pdf file.

2. QRCoder.dll

To generate QR Code passing the text to be reflected when scanned on a mobile device

3. Spire.License.dll & Spire.Pdf.dll

To Search White Colored text on the pdf file, so that the QR code can be placed at the desired location.

Out of the three tools, first two are free(open source) and the third one is used from the digital signing process.

Steps To Generate QR Code:-

Steps that are being followed while generating the QR code and stamping it onto the PDF are as follows:-

- Input PDF file is recognized to be stamped with QR Code.
- A QR code is generated containing the information of the text to be reflected after the QR code is scanned.
- The QR Code location is determined by using the “White colored text search” process
- In case the “White colored text search” doesn’t exist then default coordinates are passed.
- PDF is stamped with the QR Code.
- Final PDF containing QR Code at desired location is obtained.

Requirements for generating QR Code :

References to be used:

```
using System.Data;
using System.Data.SqlClient;
using QRCode;
using System.Configuration;
using iTextSharp.text;
using System.IO;
using iTextSharp.text.html.simpleparser;
using iTextSharp.text.pdf;
using System.Drawing;
```

Web Config Values :

```
<appSettings>
  <add key="ErrorLogSigner" value="D:\UHUDA\"/>
  <add key="TextToSearch" value="QR CODE"/>
  <add key="DefaultXCoordinate" value="163.800049"/>
  <add key="DefaultYCoordinate" value="128.911987"/>
</appSettings>
</configuration>
```

Taking Backup of Original File :

```
string msQRTexttoReturnOnScan = ("Text To Be Reflected when QR Code is scanned,QR Code Generated");
string msOriginalFilePath = "D:\\\\UHUDA\\OriginalPDF.pdf";
string msSourceFilePath = "D:\\\\UHUDA\\OriginalPDF_WithoutQRCode.pdf";
string msDestinationFilePath = "D:/UHUDA/PDFWithQRCode.pdf";
string msTextToSearchOnPdfToPlaceQRCode = System.Configuration.ConfigurationManager.AppSettings["TextToSearch"];

protected void Page_Load(object sender, EventArgs e)
{
    {
        if (!Directory.Exists("D:\\\\UHUDA"))
        {
            Directory.CreateDirectory("D:\\\\UHUDA");
        }

        if (File.Exists(msOriginalFilePath))
        {
            if (!File.Exists(msSourceFilePath))
            {
                File.Copy(msOriginalFilePath, msSourceFilePath);
                GenerateQRCode();
            }
            else
            {
                GenerateQRCode();
            }
        }
    }
}
```

Generating QR Code :

```
public void GenerateQRCode()
{
    byte[] ImageInBytes;
    QRCodeGenerator qrGenerator = new QRCodeGenerator();
    QRCodeGenerator.QRCode qrCode = qrGenerator.CreateQrCode(msQRTexttoReturnOnScan, QRCodeGenerator.ECCLLevel.Q);
    using (Bitmap bitMap = qrCode.GetGraphic(20))
    {
        using (MemoryStream ms = new MemoryStream())
        {
            bitMap.Save(ms, System.Drawing.Imaging.ImageFormat.Png);
            ImageInBytes = ms.ToArray();

            AddQRCode lobjAddQR = new AddQRCode();
            lobjAddQR.AddQRCodeToPDF(msSourceFilePath, msDestinationFilePath, msTextToSearchOnPdfToPlaceQRCode, ImageInBytes);
        }
    }
}
```

Finding Location to stamp PDF with QR Code:

```
public clsSearchText FindCoordinatedirect(string path, string wordToFind, float boxWidth, float boxHeight, PdfStamper pdfStamper, PdfReader reader, byte[] psImageBytes)
{
    clsSearchText objclsSearchText = new clsSearchText();
    {
        bool isAvailable = false;
        double x = 0, y = 0, xtemp = 0;
        float iheight = 0, iwid = 0;
        Spire.Pdf.PdfDocument doc = new Spire.Pdf.PdfDocument();
        doc.LoadFromFile(path);
        PdfTextFind[] results = null;
        QRCodeCSsharp.AddImage lobjAddImage = new QRCodeCSsharp.AddImage();

        foreach (PdfPageBase page in doc.Pages)
        {
            results = page.FindText(wordToFind).Finds;
            foreach (PdfTextFind text in results)
            {
                iheight = page.Size.Height;
                iwid = page.Size.Width;
                PointF p = text.Position;
                xtemp = iwid - text.Position.X;
                x = xtemp + (boxWidth / 2);
                y = iheight - text.Position.Y;
                int pageNo = text.SearchPageIndex + 1;

                objclsSearchText.cstXPosition = float.Parse(x.ToString());
                objclsSearchText.cstYPosition = float.Parse(y.ToString()); ;
                objclsSearchText.cstPageNum = pageNo;

                lobjAddImage.AddImageToPDF(pdfStamper, reader, psImageBytes, objclsSearchText.cstXPosition, objclsSearchText.cstYPosition, boxWidth, boxHeight, objclsSearchText.cstPageNum);
                isAvailable = true;
            }
        }
        if (isAvailable == false)
        {
            objclsSearchText.cstXPosition = float.Parse(System.Configuration.ConfigurationManager.AppSettings["DefaultXCoordinate"]);
            objclsSearchText.cstYPosition = float.Parse(System.Configuration.ConfigurationManager.AppSettings["DefaultYCoordinate"]);
            objclsSearchText.cstPageNum = 1;

            lobjAddImage.AddImageToPDF(pdfStamper, reader, psImageBytes, objclsSearchText.cstXPosition, objclsSearchText.cstYPosition, boxWidth, boxHeight, objclsSearchText.cstPageNum);
        }
    }
}
```

Stamping PDF with QR Code :

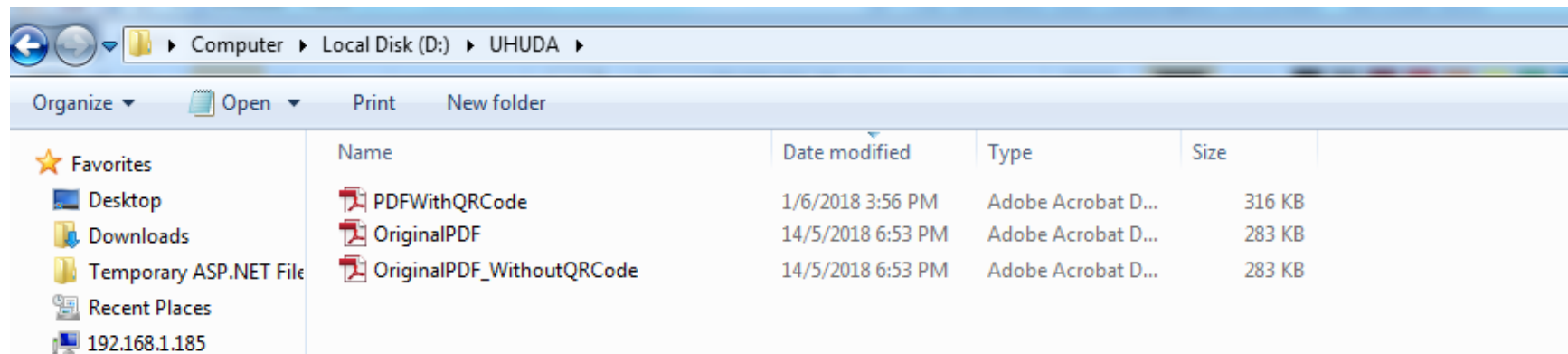
```
public void AddQRCodeToPDF(string psSrcFilePath, string psDestFilePath, string psTextToSearch, byte[] psImageBytes)
{
    ClassFiles.CustomController objCustomController = new ClassFiles.CustomController();
    byte[] pdfData = File.ReadAllBytes(psSrcFilePath);
    byte[] bytes = null;
    PdfReader reader = new PdfReader(pdfData);
    PdfReader.unethicalreading = true;
    MemoryStream ms = new MemoryStream();
    PdfStamper pdfStamper = new PdfStamper(reader, ms);
    float height = 100;
    float width = 100;

    if (psTextToSearch.Trim() != "" && psTextToSearch != null)
    {
        ClassFiles.clsSearchText objclsSearchText = new ClassFiles.clsSearchText();
        objclsSearchText = objCustomController.FindCoordinateDirect(psSrcFilePath, psTextToSearch, width, height, pdfStamper, reader, psImageBytes);
    }

    pdfStamper.Close();
    bytes = ms.ToArray();
    reader = new PdfReader(bytes);
    File.WriteAllBytes(psDestFilePath, bytes);
}
```

QR Code Samples :

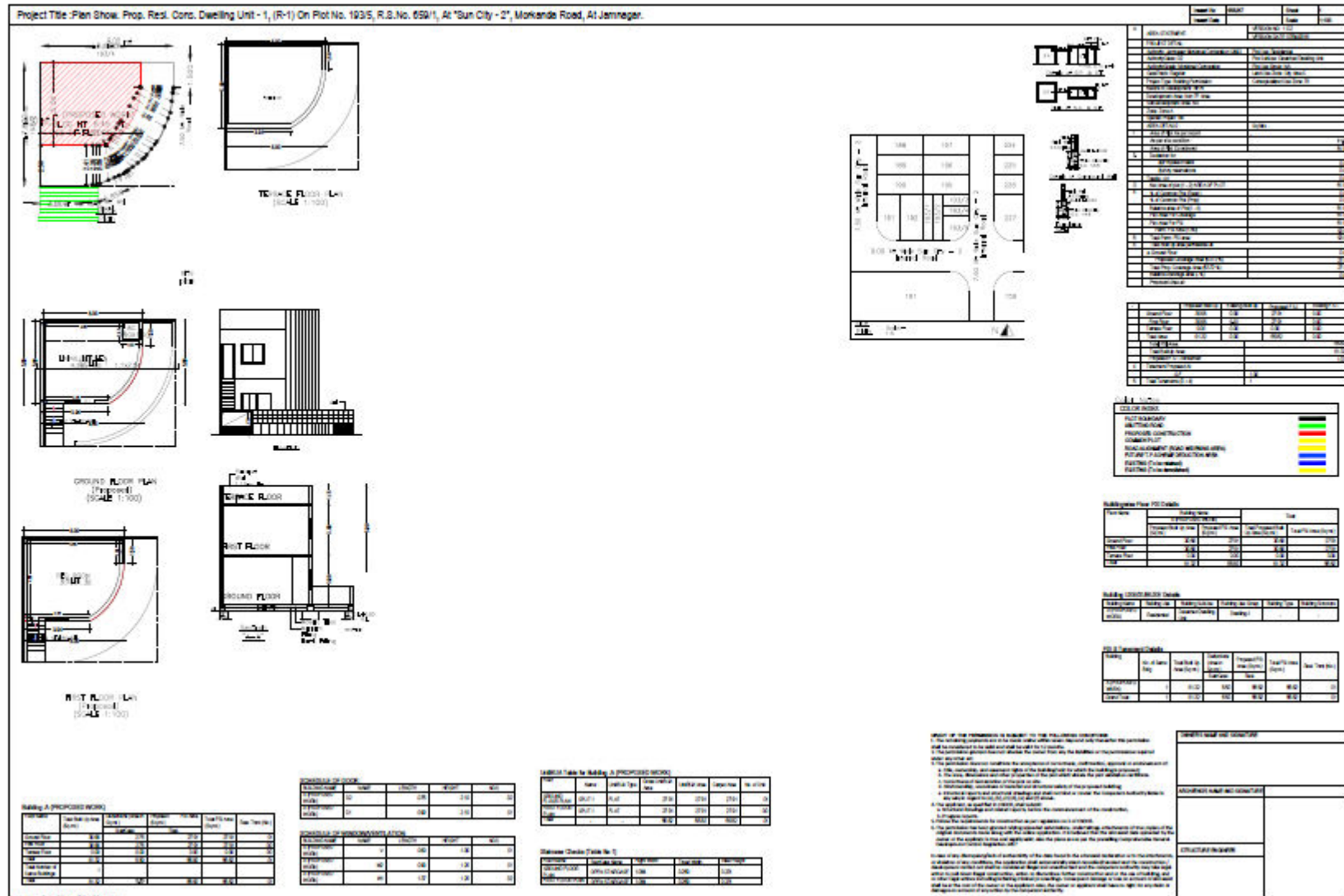
List of Original PDF and PDF with QR Code :

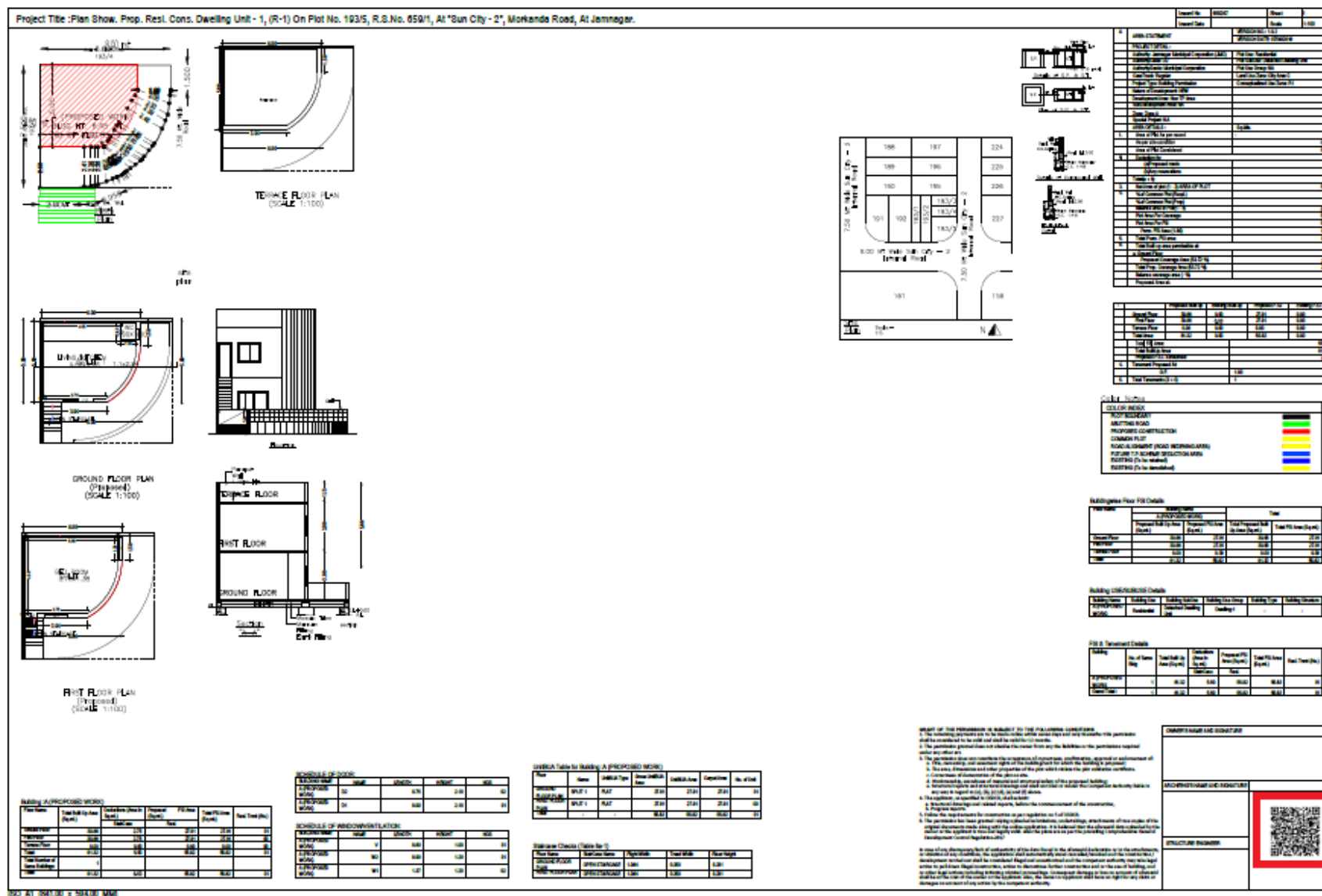


The screenshot shows a Windows File Explorer window with the address bar set to 'Computer > Local Disk (D:) > UHUDA'. The left sidebar shows 'Favorites' with links to Desktop, Downloads, Temporary ASP.NET Files, Recent Places, and 192.168.1.185. The main pane displays a table of files:

Name	Date modified	Type	Size
PDFWithQRCode	1/6/2018 3:56 PM	Adobe Acrobat D...	316 KB
OriginalPDF	14/5/2018 6:53 PM	Adobe Acrobat D...	283 KB
OriginalPDF_WithoutQRCode	14/5/2018 6:53 PM	Adobe Acrobat D...	283 KB

Original PDF:



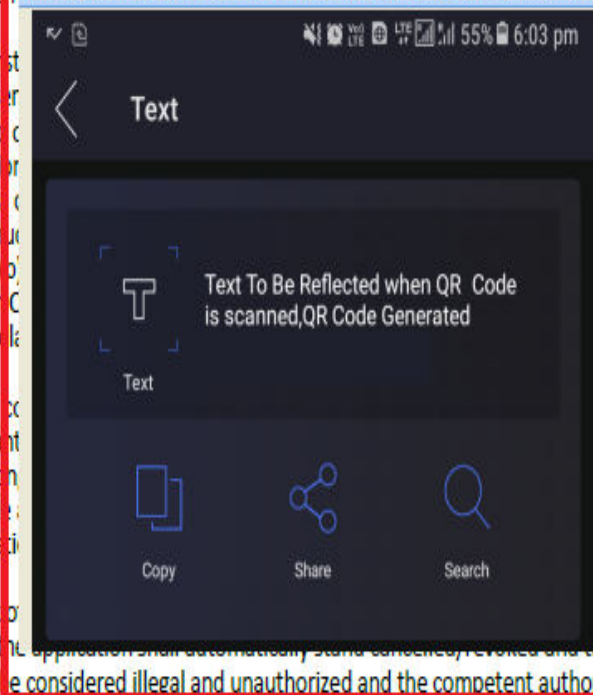


Text received when QR code is scanned on Mobile Device :

GRANT OF THE PERMISSION IS SUBJECT TO THE FOLLOWING CONDITIONS

1. The remaining payments are to be made online within seven days and only thereafter this permission shall be considered to be valid and shall be valid for 12 months.
2. The permission granted does not absolve the owner from any the liabilities or the permissions required under any other act.
3. The permission does not consist of:
 - a. Title, ownership, and easement.
 - b. The area, dimensions and location.
 - c. Correctness of demarcation.
 - d. Workmanship, soundness.
 - e. Structural reports and structural drawings.
4. The applicant, as specified in the following:
 - a. Structural drawings and related documents.
 - b. Progress reports.
5. Follow the requirements for the grant of permission.
6. The permission has been granted on the basis of the true copies of the original documents made along with the application and the owner or the applicant is true and correct.

In case of any discrepancy/lack of information or violation of any conditions, the application shall automatically stand cancelled, revised and the construction / development carried out shall be considered illegal and unauthorized and the competent authority may take legal action to pull down illegal construction, action to discontinue further construction and or the use of building, and or other legal actions including initiating criminal proceedings. Consequent damage or loss on account of aforesaid shall be at the cost of the owner or the applicant. Also, the owner or applicant shall have no right for any claim or damages on account of any action by the competent authority.



OWNER'S NAME AND SIGNATURE

ARCH/ENG'S NAME AND SIGNATURE

STRUCTURE ENGINEER

