

I send you CaptureScreenDll reference with PDF option.

```
#include <CtrlLib/CtrlLib.h>
#include <plugin/bmp/bmp.h>
#include <plugin/jpg/jpg.h>
#include <plugin/gif/gif.h>
#include <plugin/tif/tif.h>
#include <Report/Report.h>
#include <PdfDraw/PdfDraw.h>
```

```
using namespace Upp;
```

```
Image ScreenShot(int x, int y, int cx, int cy)
```

```
{
    HDC dcScreen = CreateDC("DISPLAY", NULL, NULL, NULL);
```

```
    RGBA *pixels;
```

```
    Buffer<byte> data;
    data.Alloc(sizeof(BITMAPINFOHEADER) + sizeof(RGBQUAD)*256);
    BITMAPINFOHEADER *hi = (BITMAPINFOHEADER *) ~data;;
    memset(hi, 0, sizeof(BITMAPINFOHEADER));
    hi->biSize = sizeof(BITMAPINFOHEADER);
    hi->biPlanes = 1;
    hi->biBitCount = 32;
    hi->biCompression = BI_RGB;
    hi->biSizeImage = 0;
    hi->biClrUsed = 0;
    hi->biClrImportant = 0;
    hi->biWidth = cx;
    hi->biHeight = -cy;
```

```
    HBITMAP hbmp = CreateDIBSection(dcScreen, (BITMAPINFO *)hi, DIB_RGB_COLORS, (void
**)&pixels, NULL, 0);
```

```
    HDC dcMem = ::CreateCompatibleDC(dcScreen);
    HBITMAP hbmpOld = (HBITMAP) ::SelectObject(dcMem, hbmp);
    HDC hdcCompatible = CreateCompatibleDC(dcScreen);
    ::BitBlt(dcMem, 0, 0, cx, cy, dcScreen, x, y, SRCCOPY);
```

```
    ImageBuffer ib(cx, cy);
    memcpy(~ib, pixels, cx * cy * sizeof(RGBA));
```

```
    ::DeleteObject(::SelectObject(dcMem, hbmpOld));
    ::DeleteDC(dcMem);
```

```

::DeleteDC(dcScreen);

return ib;
};

void PDFEncoder(String filename, Image m)
{
    FileSel fs;
    Report r;
    r.DrawImage(0,0,Rescale(m,r.GetSize()));
    fs.Set(filename);
    SaveFile(~fs, UPP::Pdf(r));
};

extern "C" __declspec(dllexport) int capture_screen(int x, int y, int cx, int cy, char *filename) {
    Image m = ScreenShot(x, y, cx, cy);
    String ext = ToLower(GetFileExt(filename));
    if(ext == ".png")
        PNGEncoder().SaveFile(filename, m);
    if(ext == ".bmp")
        BMPEncoder().Bpp(32).SaveFile(filename, m);
    if(ext == ".jpg")
        JPGEncoder().SaveFile(filename, m);
    if(ext == ".gif")
        GIFEncoder().SaveFile(filename, m);
    if(ext == ".tif" || ext == ".tiff")
        TIFEncoder().SaveFile(filename, m);
    if(ext == ".pdf")
        PDFEncoder(filename, m);
    return 0;
}

DLL_APP_MAIN {
}

```

File Attachments

1) [CaptureScreenDll.upp](#), downloaded 434 times
