



XIX Edizione, 2020

Modernizzare app VCL

Esempi pratici, componenti Win10
Temi dark/light - IconsFont e SVGIcons
High-DPI... e l'uso del Windows Store



CARLO BARAZZETTA

ETHEA

SOCIO

FONDATORE

E RESPONSABILE

PROGETTI IT

 **DelphiDay**
italian conference

ONLINE
EDITION

 **wintech**
italia

WWW.ETHEA.IT



GITHUB PROJECTS



github.com/EtheaDev



AGENDA

- Perché è necessario modernizzare le app VCL?
- Come IconFontImageList e SVGIconImageList ci aiutano
- Tema chiaro/scuro, va di moda: (TVCLThemeSelector)
- Il supporto High-DPI: eventi e gestione avanzata
- Il ruolo dei Font e possibilità di personalizzazione
- I componenti per Windows 10: panoramica e fix
- Tipologie di Resize/Rescaling
- Windows Store: un'opportunità...
- Tecniche avanzate per il deploy su Windows Store

MODERNIZZAZIONE DI APPLICAZIONI VCL

Materiale on-line

- IconFontsImageList:
 - <https://github.com/EtheaDev/IconFontsImageList>
- SVGIconImageList:
 - <https://github.com/EtheaDev/SVGIconImageList>
- VCLThemeSelector e Demo High-DPI:
 - <https://github.com/EtheaDev/VCLThemeSelector>
 - Form “elegante” per selezionare il tema della app
 - Demo completa con i concetti di modernizzazione e supporto High-DPI.

VCLThemeSelector: necessario almeno Delphi XE3. ModernAppDemo: necessario almeno Delphi 10.1.

Perché è necessario modernizzare?

- **Problemi di App esistenti:**
 - Le app VCL sono nate decine di anni fa (default Font Tahoma)
 - A volte utilizzano componenti di terze parti non più aggiornati
 - Alcuni paradigmi di UI sono vecchi: MainMenu, CheckBox?
- **Problemi con le nuove App:**
 - Supporto High-DPI di default
 - Nuovi componenti (es. FlowPanel) e componenti per Windows 10
 - Dark Mode / Light Mode: non è solo una moda
- **Problemi hardware:**
 - Schermi ad alta risoluzione 2K o 4K

Da Windows 7 a Windows 10...

Tema Dark, Material design Icons, Hamburger menu, High-DPI

Basket Scouting Pro - D:\ethea\BasketScouting\trunk\Samples\ITA\AllStarGame2007\test_solo_per_immagini_helpSerie_C_fine2q.bsgame

IMBECILLI.IT ALL STAR GIR.B: Giocatori in campo

Player	TL	2P	3P	RO	FF	SD	0
Elena	4	0	3	1	0	0	0
Martini	0	0	0	0	0	0	0
Sustinente	0	0	0	0	0	0	0

ETHEA ALL STAR GIR.A: Giocatori in campo

Player	TL	2P	3P	RO	FF	SD	0
Paola	9	1	1	1	0	0	0
Romeri	0	0	0	0	0	0	0
Sondrio	0	0	0	0	0	0	0

Mapa di tiro

Ethea All Star Gir.A

Player	TL	2P	3P	RO	FF	SD	0
ETH	5	10	1	2	0	0	0
1	0	4	17	1	2	4	1
40.00	%	55.6	37.0	50.0	PPR	1	SS
0							PT
28							

Coloni differenti

Stampa

Chiudi (Esc)

Aiuto

7 1 Caterina Martinazzoli (Crema) 4

8 0 Elena Tosì (Pavia 94) 4

9 0 Erica Morselli (Sustinente) 0

10 0 Giulia Cigoli (Cremona) 3

11 2 Silvia Gandini (Ghost Pavia) 0

12 0 Caterina Belli (Pavia) 2

2° Tempo 00:00

IMB 35

Basket Scouting 6.0.0 - (beta00)

ETHEA LOMBARDIA EST SERIE A-B: on court

Player	FT	FG	3P	OREB	FF	BLK	0
Cielo	4	0	0	0	0	0	0
De Luca	0	0	0	0	0	0	0
Varesè	0	0	0	0	0	0	0

DREAM TEAM LOMBARDIA OVEST SERIE A-B: on court

Player	FT	FG	3P	OREB	FF	BLK	0
Ludovica	4	0	0	0	3	0	0
Ferrì	0	0	0	0	0	0	0
Biassono	0	0	0	0	0	0	0

Shooting Map

ETHEA

Player	FT	FG	3P	OREB	FF	BLK	0
ETH	11	17	8	0	14	0	0
2	0	12	31	12	0	7	4
20.02	%	47.8	35.4	40.0	STL	0	PTS
49							

4° Tempo - game in progress

00:00

69 71 DRE

3 6 0 0 0 0 0 0

Team Pts: 00:00

Team Rebs: 00:00

DREAM TEAM LOMBARDIA OVEST SERIE A-B Substitutes

Player	Score	
10 0	Giuseppina Nucci (Cantù)	7
11 2	Fedra Cocci (Pool Bustese)	2
12 1	Agnese Ricci (Costa Masnaga)	11
13 0	Donata Fiorentini (Varese)	13
14 0	Irina Buccho (Cucciago)	0

HIGH-DPI: la differenza si vede!

Applicazione senza High-DPI

DPI virtualization (da Windows Vista)

Species No	Graphic
90030	
Category	
Snapper	
Common_Name	
Red Emperor	
Species Name	
Lutjanus sebae	
Length (cm)	
60	
Length_In	
23,6220472440945	
Notes	

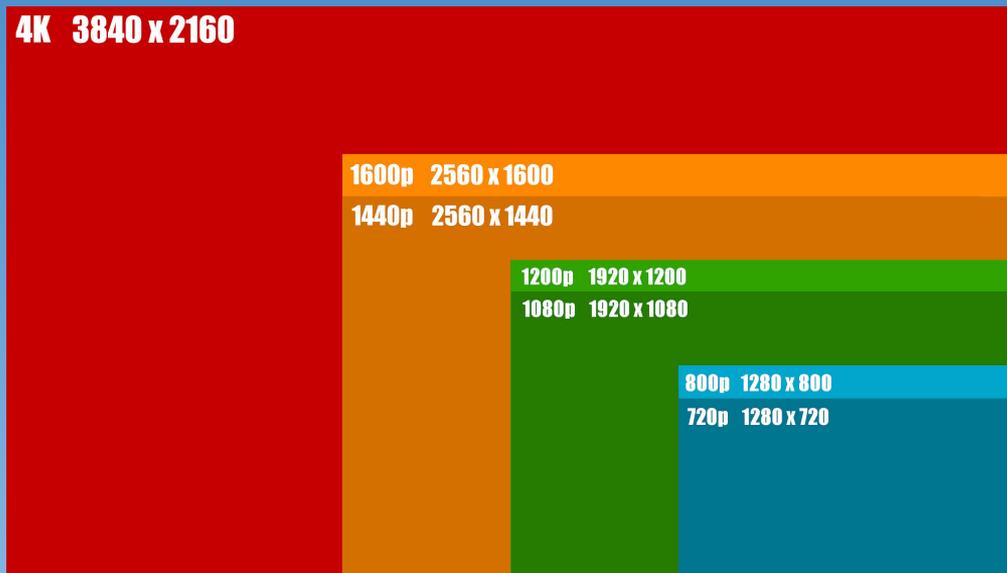
Applicazione con High-DPI

Per monitor DPI-scaling (da Windows 8.1)

Species No	Graphic
90030	
Category	
Snapper	
Common_Name	
Red Emperor	
Species Name	
Lutjanus sebae	
Length (cm)	
60	
Length_In	
23,6220472440945	
Notes	

Non solo pixel... anche dim. testo

Diverse risoluzioni, ma anche diversa % del Font, cioè DPI (DotPerInch) = PPI (PixelsPerInch)



con gli schermi 2K su notebook da 14/15 pollici e sugli schermi 4K è facile trovare impostazioni di 125% o 150% o anche 200%.

Schermo

Ridimensionamento e layout

Alcune app non rispondono alle modifiche del ridimensionamento finché non le chiudi e le riapri.

Modifica la dimensione di testo, app e altri elementi

150% (scelta consigliata) ▾

Impostazioni ridimensionamento avanzate

Risoluzione dello schermo

3840 x 2160 (scelta consigliata) ▾

Orientamento dello schermo

Orizzontale ▾

Più schermi

Più schermi

Estendi questi schermi ▾

Imposta come schermo principale

2 problemi: icone e font

- I componenti scalano (forse, non benissimo...) ma le icone?
 - Le ImageList classiche non forniscono alcun supporto
 - Da Delphi 10.3 esiste VirtualImageList e ImageCollection... e prima di 10.3?
 - Diffusione di 2 standard:
 - material design
 - SVG icons
- Il font della nostra applicazione
 - Si adatta al font del S.O. (Segoe UI)
 - Permettiamo all'utente di personalizzarlo?

Come IconFontsImageList ci aiuta...

- Progetto: github.com/EtheaDev/IconFontsImageList
 - Funziona “come” una VirtualImageList!
 - Disponibile gratuitamente (da Delphi 7 a 10.4)
 - Disponibilità di molte icone già pronte
 - esempio: github.com/Templarian/MaterialDesign-Font
 - Possibilità di cambiare colore e dimensione facilmente
 - supporto alle icone “surrogate pair”
 - CharMap e IconManager con metadati sulle icone (name)
 - E' disponibile anche una versione per FMX (Win-Android-iOS)
 - Wiki completa: github.com/EtheaDev/IconFontsImageList/wiki



SVGIconImageList: il gemello

- Progetto: github.com/EtheaDev/SVGIconImageList
 - Funziona “come” una VirtualImageList!
 - Disponibile gratuitamente (da DXE6 a 10.4)
 - Disponibilità di molte icone già pronte free e a pagamento
 - esempio: github.com/PapyrusDevelopmentTeam/papyrus-icon-theme
 - Possibilità di cambiare dimensione/colore/scale di grigio facilmente
 - esiste anche il TSVGIconImage per visualizzare le immagini SVG
 - E' disponibile anche una versione per FMX (solo Windows platform)
- Libreria grafica: DelphiSVG di Martin Walter (2008)... applicati alcuni bug-fix
 - Progetto inattivo: <https://development.mwcs.de/svgimage.html>



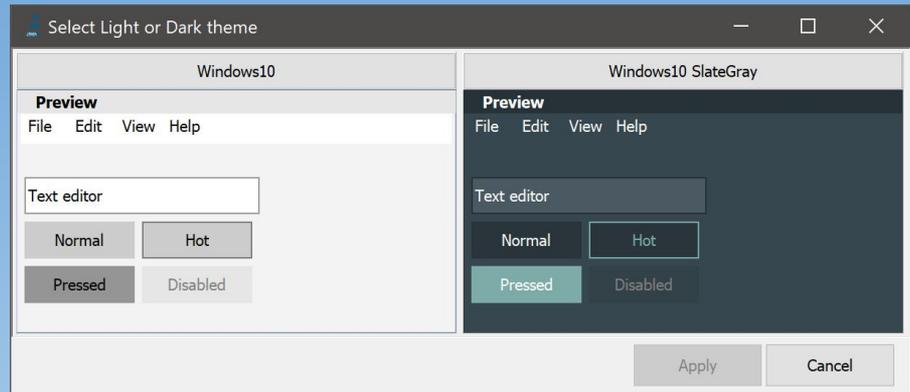
Modernizzare con i VCLStyles

- Sfruttiamo gli stili della VCL (disponibili da XE2)
 - Come si abilitano e si aggiungono
 - Disponibilità di nuovi stili in GetIt
 - Proprietà: StyleElements (per “disattivarli”)
- Il “non stile” Windows
 - Attenzione ai componenti “owner-draw”
 - Attenzione agli attributi Ctrl3D, Bevel, Border, ecc...
- Uso dei colori:
 - TStyleManager.ActiveStyle.GetStyleColor, GetStyleFontColor, GetSystemColor, GetElementColor...

Tema chiaro/scuro

- Sfruttiamo gli stili della VCL
 - Attenzione alla proprietà StyleElements (per disattivarli).
- Usiamo il preview-selector TVCLThemeSelector per semplificare la scelta
 - Escludere o meno il “non stile” Windows
 - Autoridimensionamento (MaxCols e MaxRows)
 - Come si integra (provare il VCLThemeSelectorLauncher)
 - Basato su

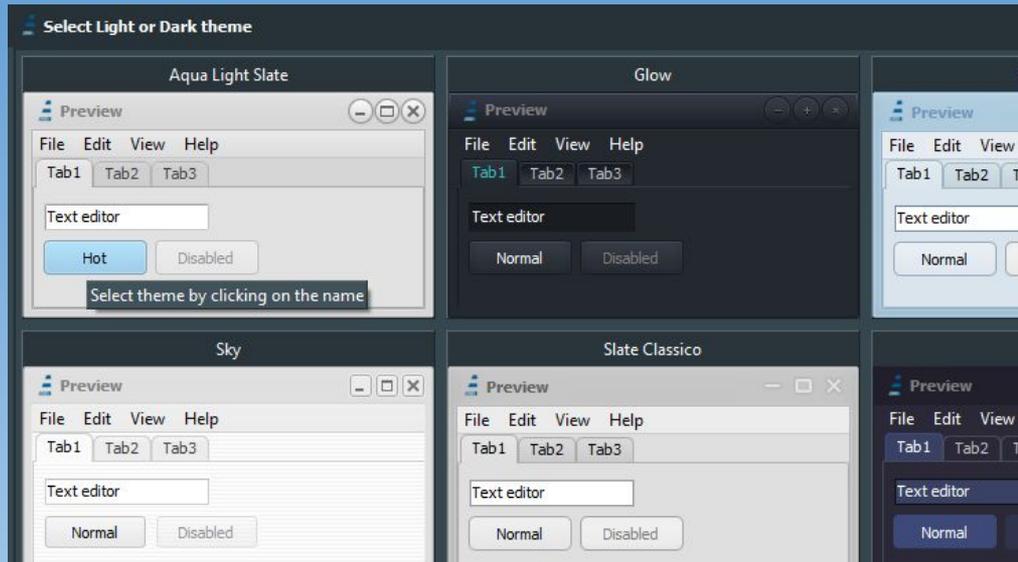
VCLStylePreview
della libreria VCLStyleUtils
ma con supporto High-DPI



VCLThemeSelector con D10.4

- Novità 10.4: VCLTheme Selector sfrutta “per control-style”
 - E’ possibile utilizzare più stili contemporaneamente
- Il preview/selector TVCLThemeSelector è stato ridisegnato
 - Utilizzo di una form di preview con oggetti “reali” e “interattivi”
 - Non necessita

più di
VCLStylePreview



Il supporto High-DPI

- **Come funziona? Come si attiva in Delphi?**
 - **Documentazione MS:**
<https://docs.microsoft.com/it-it/windows/win32/hidpi/high-dpi-desktop-application-development-on-windows?redirectedfrom=MSDN>
 - **Documentazione Embarcadero:**
 - <https://community.idera.com/developer-tools/b/blog/posts/how-to-guide-upgrading-your-delphi-vcl-applications-to-support-4k-displays>
- **Supporti differenti in base alla versione di Delphi (consigliato D10.3)**
 - **Delphi fa quasi tutto il lavoro in automatico se:**
 - `Form.Scaled = True, Form.ParentFont = False (Form.PixelsPerInch=96)`
 - **Attenzione alle impostazioni “autosize”**
- **In Windows parliamo di “logical DPI” che non corrisponde a quella fisica!**

Il DPI “logico” in Windows

Il valore DPI in Windows è “logico” e non ha a che fare con il DPI fisico del Display. Esso definisce quanto “largo” un testo è mostrato rispetto al valore costante di **96** (Form.PixelsPerInch default), che corrisponde al 100% nelle impostazioni.

Es. **144** DPI (= 96×1.5) significa che gli elementi sullo schermo (non solo il font) appariranno il 50% più grandi del “normale” (impostazione 150%).

Es. **192** DPI (= 96×2) è l'impostazione di uno schermo 4K al 200% che mostra sullo schermo le stesse “proporzioni” di un comune 2K al 100%.

Schermo

Ridimensionamento e layout

Alcune app non rispondono alle modifiche del ridimensionamento finché non le chiudi e le riapri.

Modifica la dimensione di testo, app e altri elementi

150% (scelta consigliata)

Impostazioni ridimensionamento avanzate

Risoluzione dello schermo

3840 x 2160 (scelta consigliata)

Orientamento dello schermo

Orizzontale

Più schermi

Più schermi

Estendi questi schermi

Imposta come schermo principale

VCLStyleUtils e oltre...

- VCLStyleUtile è una libreria molto utile per fixare alcuni comportamenti anomali degli stili nelle precedenti versioni di Delphi.
 - github.com/RRUZ/vcl-styles-utils
 - Per fixare Menu, Dialogs, Components, TaskDialog
 - Soprattutto con vecchie versioni di Delphi (da XE2)
 - Non c'è ancora supporto per D10.4
- Con Delphi invece 10.4 sono stati fixati diversi componenti per il supporto High-DPI Styles, oltre agli stili con la grafica ad alta-risoluzione.

TFont panoramica e utilizzo

- **Proprietà Size e Height:**
 - Size è il “corpo” che è disponibile nel font
 - Height è la dimensione in Pixel (con segno negativo)
 - PixelsPerInch è la proprietà DPI dello schermo principale
 - N.B. usare sempre Height per i calcoli (maggior precisione)
- **Ruolo di Application.DefaultFont e di Screen.IconFont**
- **Vediamo la demo TestDPI per capire come funziona e come gestirli**
- **I font di sistema: Screen.MenuFont, Screen.MessageFont, Screen.HintFont, Screen.CaptionFont**

Esempio TestDPI

```
procedure TMainForm.Loaded;
begin
    //Very important for HighDPI: on Main Screen ParentFont must be always be False
    ParentFont := False;
    //Acquire system font and size (eg. for windows 10 Segoe UI and 14 at 96 DPI) but without using Assign!
    Font.Name := Screen.IconFont.Name;
    //If you want to use system font Height:
    Font.Height := Muldiv(Screen.IconFont.Height, 96, Screen.IconFont.PixelsPerInch);
    (* //Sample assign Font by user preferences:
    Font.Name := 'Century Gothic';
    Font.Color := clBlue;
    Font.Height := -14; *)
    inherited;
    //For Child Forms with ParentFont = True
    UpdateDefaultAndSystemFonts;
    //Test build run-time components
    //BuildLabels;
end;
```

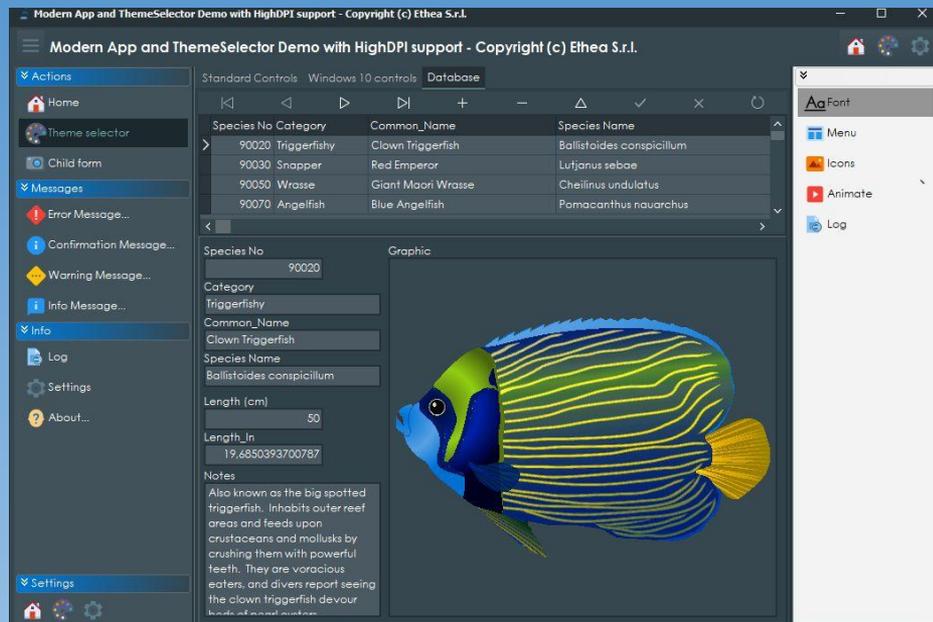
I componenti per Windows 10

- Da Delphi 10 Seattle sono stati introdotti alcuni componenti nuovi:
 - TRelativePanel, TSplitview*, TToggleSwitch*, TActivityIndicator, TSearchBox*
- Da Delphi 10.1.2 sono stati introdotti:
 - TCalendarView*, TCalendarPicker*
- Da Delphi 10.2.2 sono stati introdotti:
 - TDatePicker*, TTimePicker*

*Diamo uno sguardo alla ModernAppDemo per vederli all'opera e come vanno fixati (la demo è compatibile da Delphi 10.1 in poi).

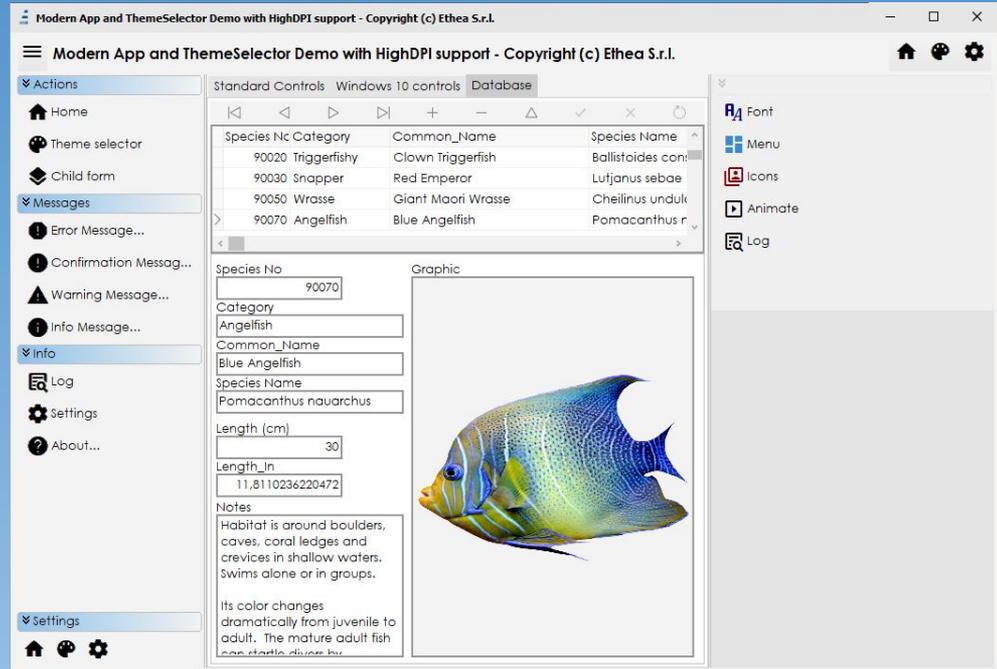
Panoramica ModernAppDemo

- Utilizzo di IconFontsImageList (multicolor e styled) e di SVGIconImageList
- Utilizzo TSplitView e TCategoryButton
- Uso VCLStyles con VCLThemeSelector
- Menu Settings “senza” stile
- Utilizzo di controlli Windows 10 e fix
- Esempio Immagini Hi-Res - Biolife ;-)
- Personalizzazione Font Name, Height
- System fonts (Menu, Hint, Messages)
- VCLStyleUtils (TaskDialog, Dialog)
- Child Form con ParentFont
- High-DPI e cambio risoluzione al volo



Panoramica ModernAppDemo

- Provare queste azioni:
 - Resize e change del font
 - Cambiare il tema
 - Switch tra IconFonts e SVGFont
 - Animazioni del main menu
 - Message dialogs
- Osservare nel codice
 - \$ifdef per le differenze tra versioni di Delphi
 - Impostazioni System fonts
- Child Form con ParentFont
- High-DPI e cambio risoluzione al volo



High-DPI: Personalizzare il font

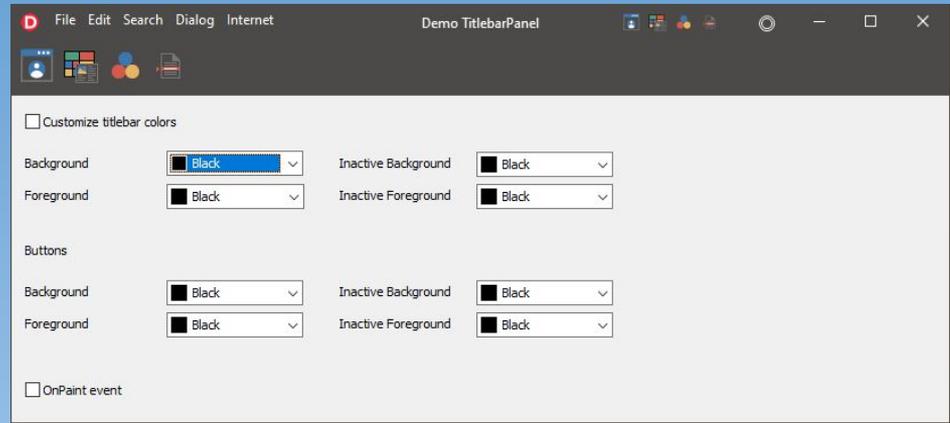
- **N.B. ParentFont True o False cambia il comportamento:**
 - a. Con False viene scalato automaticamente il Font della Form
 - b. Con True viene assegnato alla Form Application.DefaultFont
- **2 alternative:**
 - a. Main e ChildForm con ParentFont a False:
 - VCL fa il resize
 - Riassegnazione Font personalizzato in ogni form (o tramite una Child form di base)
 - b. MainForm con ParentFont a False e Child forms con ParentFont a True:
 - VCL fa il resize solo della Main
 - La Child “eredita” Application.DefaultFont
 - Vantaggio: una sola assegnazione
 - Svantaggio: impostare ParentFont a true nei dfm di tutte le form (non è il default)
- **Suggerimento: utilizzare una form di base per gestire le casistiche**

High-DPI: gli eventi da utilizzare

- **Screen.PixelsPerInch e Form.Monitor.PixelsPerInch**
 - Screen.PixelsPerInch è il monitor principale
 - Font.Monitor.PixelsPerInch è legato al monitor in cui la form è visibile
- **OnAfterMonitorDpiChanged**
 - Quando si cambia monitor o quando si cambiano le impostazioni
 - Quando si lancia una ChildForm da un monitor diverso dal primario
- **ScaleForPPI**
 - Metodo della form che scala i componenti: utile per fixare componenti VCL o di terze parti che non scalano correttamente
- **ChangeScale**
 - Metodo che i componenti ricevono per scalare: utile per “sistemare” il comportamento internamente al componente costruendosi propri componenti.

Il nuovo TitleBarPanel (solo D10.4)

- Sostituisce il titolo della finestra
 - Permette di avere qualsiasi controllo sul titolo
 - Permette di personalizzare i colori
 - Menu di sistema e pulsanti di sistema replicati
 - Presente nelle Demo 10.4
 - Ancora qualche “problema” con il supporto High-DPI
 - Non usato nella demo: solo a titolo informativo



Resize in base allo schermo

- Considerare l'impostazione High-DPI non sempre è la soluzione
 - Dipende soprattutto se è pensata per funzionare a tutto schermo (es. su Tablet Windows)
- Esempio di BasketScouting:
 - Il fattore di scala va ricalcolato solo in base alla dimensione dello schermo
 - L'interfaccia utente scala diversamente anche in base al contenuto/contesto



DEPLOY DI APP SU WINDOWS STORE

Perché utilizzare il Microsoft Store



<https://community.idera.com/developer-tools/b/blog/posts/deploying-windows-10-apps-through-the-microsoft-store-1863306656>

I vantaggi dell'aggiunta di una applicazione Delphi al Microsoft Store sono numerosi:

- **Rilevabilità:** l'app verrà elencata in una directory mondiale e può essere scoperta da persone che non avresti mai raggiunto, potenzialmente in tutto il mondo.
- **Facile installazione e disinstallazione:** l'installazione e la disinstallazione delle app di Windows ha sempre avuto delle difficoltà: hai bisogno dei diritti di amministratore, ogni programma di installazione è diverso dall'altro, devi creare script di installazione e, disinstallando l'app, lasci sempre tracce che ingombreranno il tuo computer. Con Microsoft Store, l'installazione e la disinstallazione sono molto semplici e veloci!
- **Sicurezza:** le app scaricate da Microsoft Store sono garantite come sicure: nessun virus o danno al tuo computer. Le app vengono eseguite in una sandbox e non danneggiano né te né il tuo computer (**non proprio vero...**)
- **Monetizzazione:** puoi utilizzare diversi modi per monetizzare la tua applicazione; addebitare un prezzo per scaricare l'applicazione, aggiungere acquisti in-app all'applicazione, aggiungere annunci all'applicazione o persino utilizzare un modello basato su abbonamento per l'applicazione. Lascia che lo Store faccia tutto questo per te!

Registrazione e configurazione

- Creare un account sviluppatore (singolo a azienda)
 - <https://developer.microsoft.com/en-us/store/register/>
- Singolo sviluppatore: 19 \$ una tantum
- Azienda: 99 \$ una tantum
 - <https://docs.microsoft.com/it-it/windows/uwp/publish/account-types-locations-and-fees>
- Attenzione: Impostare la lingua inglese durante il processo di registrazione, per riuscire a completare il processo!

Centro per i Partner

Programmi per gli sviluppatori



Windows e Xbox

Registrati come sviluppatore di app per inviare app e giochi ai marketplace Microsoft.

[Per iniziare](#)



Marketplace commerciale

Pubblica offerte di software e servizi nel marketplace commerciale di Microsoft per raggiungere milioni di clienti aziendali e di enti pubblici tramite esperienze all'interno del prodotto, Azure Marketplace, Microsoft AppSource, il programma Cloud Solution Provider e le vendite Microsoft.

[Per iniziare](#)



Office Store

Sviluppa app o componenti aggiuntivi che supportano o migliorano la familiare esperienza di Office, SharePoint e Office 365.

[Per iniziare](#)



Hardware

Registra la tua azienda per certificare e distribuire driver e dispositivi compatibili con Windows.

[Per iniziare](#)



Microsoft Collaborate

Registrati per partecipare ad anteprime, versioni non definitive, impegni di collaborazione e co-progettazione Microsoft per partner e clienti esterni. Questo include definizione degli impegni, scambio di



Applicazioni desktop di Windows

Accedi ai report di telemetria e analisi sulle tue applicazioni desktop firmate con certificato.

[Per iniziare](#)

Selezionare
“Windows e Xbox”,
non Applicazioni
desktop!

Menu Windows e Xbox

The screenshot shows the Microsoft Partner Center interface. At the top, it says 'Centro per i partner Microsoft Ethea S.r.l.' with a search bar. The left sidebar contains a navigation menu with 'Home page', 'Directory: carlo.barazzetta@ethea.it', and a list of categories: 'Windows e Xbox' (highlighted with a red box), 'Panoramica', 'Prodotti', 'Attrai', 'Coinvolgi', 'Monetizza', and 'Analizza'. The main content area is titled 'Panoramica' and features a 'Crea una nuova app' button (also highlighted with a red box). Below this, there is a card for 'Basket Scouting' with a logo, the text 'App - In attesa del rilascio', and a 'Continua invio' link. To the right of the card, it shows 'Incluso: 1 componenti aggiuntivi' and 'Mercati: 241'. At the bottom of the card, there is a note: 'I valori precedenti indicano il feedback e gli acquisti per l'intero periodo, le recensioni e valutazioni negli ultimi 30 giorni e gli arresti anomali nelle ultime 72 ore (dati non...)'.

Creare una nuova app...

Riservare il nome!

Crea la tua app riservando un nome

Dopo aver riservato un nome, verrà eseguito il provisioning della tua app per servizi come le notifiche push e potrai iniziare a definire gli add-on.

Assicurarsi di disporre dei diritti per usare qualsiasi nome prenotato. È necessario inviare l'app al Microsoft Store entro tre mesi oppure la prenotazione del nome decadrà. [Altre informazioni](#)

Nome *

MiaApp

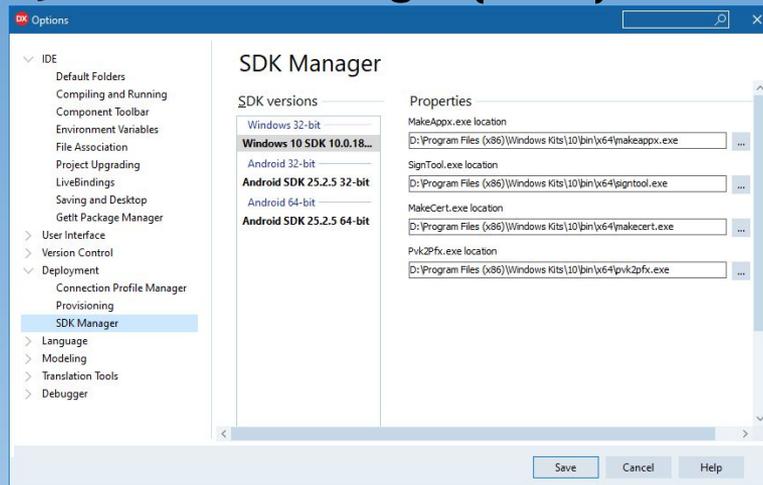
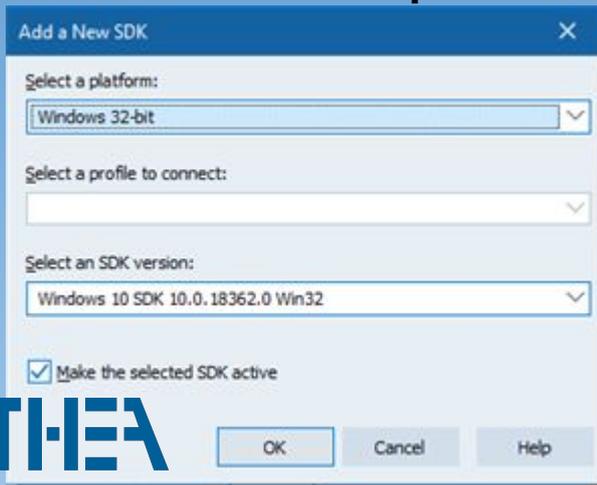
✓ Verifica disponibilità

Riserva nome del prodotto Annulla

Riservare il nome e pubblicare la app entro 3 mesi dalla prenotazione!

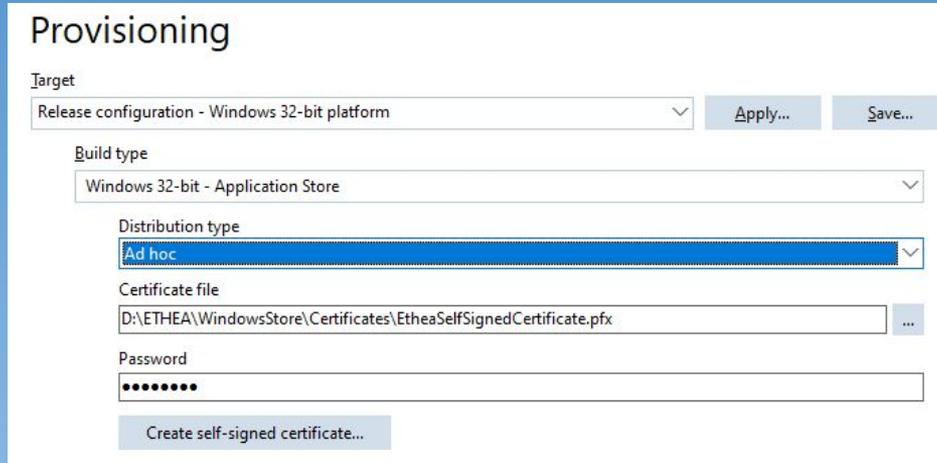
Installazione SDK

- Installazione SDK nell'IDE (Utilizzare Delphi 10.3 Rio)
 - Scaricare e installare il Windows 10 SDK (winsdksetup.exe)
 - <https://developer.microsoft.com/it-it/windows/downloads/windows-10-sdk/>
 - Aprire SDK Manager in Delphi 10.3 IDE
 - Tools/Options/Deployment/SDK Manager (Add...)

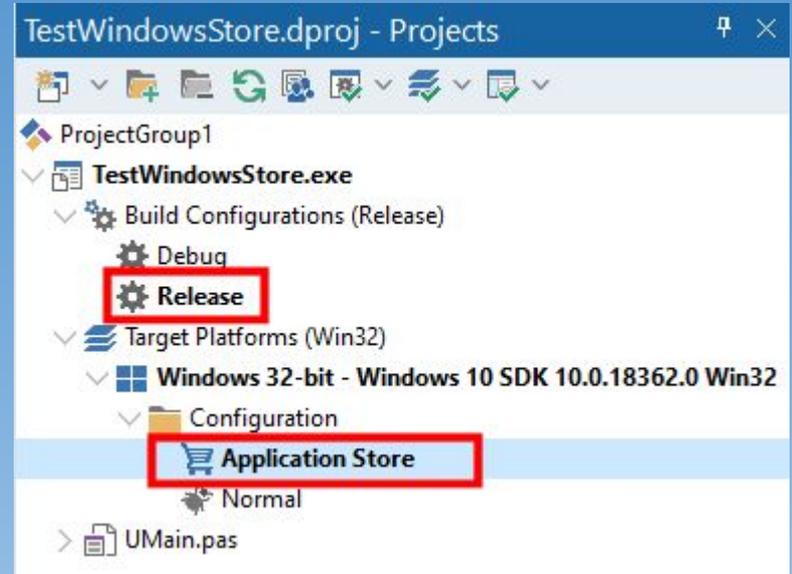


Build options del pacchetto AdHoc

- Project/Options/Provisioning:
 - Ad hoc

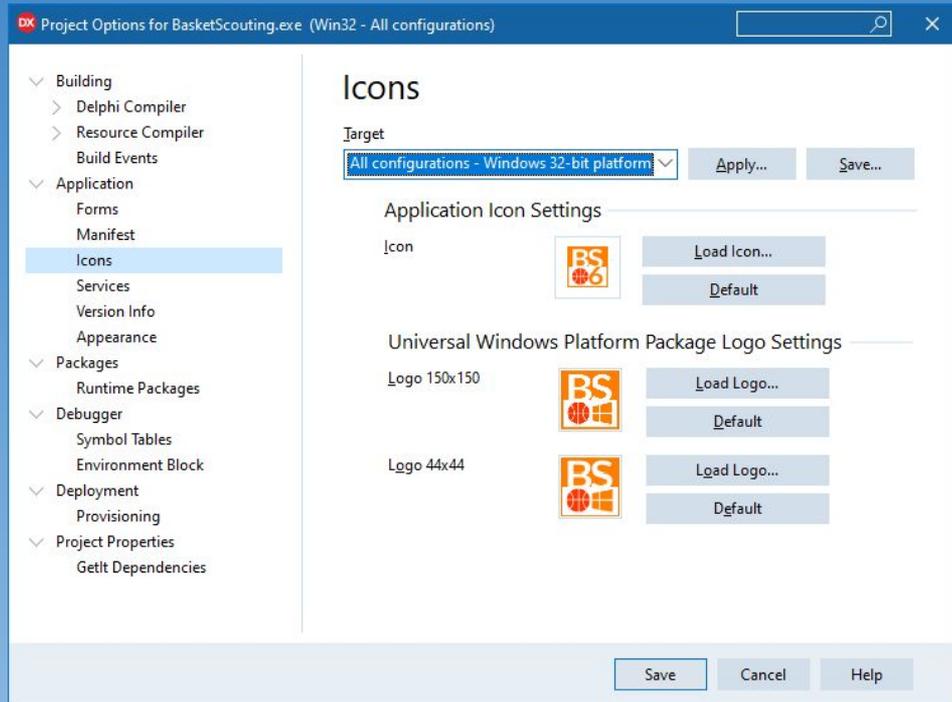


- Build options:
 - Release - Application Store



Gestione Icone

- Icona applicazione usare .ico multires 16,32,44,48,64,128,150,256
- <https://icoconvert.com/>
- Icone per il Windows Store
 - Logo 150x150
 - Logo 44x44



Provisioning: Build Type e Distrib.Type

- **Project Options/Deployment/Provisioning**
 - **Build type: Windows 32 bit - Application Store**
 - **Distribution Type:**
 - **AdHoc (richiede certificato Self-Signed)**
 - **Store (per il deploy su Windows Store)**
 - **Creare il certificato self-signed**
- **Processo di Deploy AdHoc:**
 - **Build (release) - MakeAppx - Sign (con Self-signed certificate)**
- **Processo di Deploy Store:**
 - **Build (release) - MakeAppx - la firma la appone lo store**

Provisioning: Creare il certificato self-signed

DX Create a new Certificate ✕

New Certificate

A certificate is a file to sign Windows applications. Please specify a filename and subject for your certificate. The specified filename must not exist, and have .pfx extension.



File:
D:\ETHEA\EtheaMicrosoftStore\EtheaSelfSignedCertificate.pfx ...

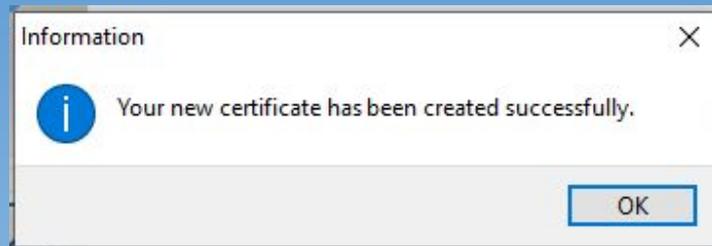
Subject:
CN=Ethea S.r.l., C=Italy

Password:
●●●●●●

ETHEA

<< Back Next >> Finish Cancel Help

Premere finish...



Provisioning: certificato creato

DX Options

> IDE
> User Interface
> Version Control
v Deployment
 Connection Profile Manager
 Provisioning
 SDK Manager
> Language
> Modeling
> Translation Tools
> Debugger

Provisioning

Build Type
Windows 32-bit - Application Store

Distribution type
Ad hoc

Certificate file
D:\ETHEA\EtheaMicrosoftStore\EtheaSelfSignedCertificate.pfx

Password
.....

Create self-signed certificate...

Certificate info

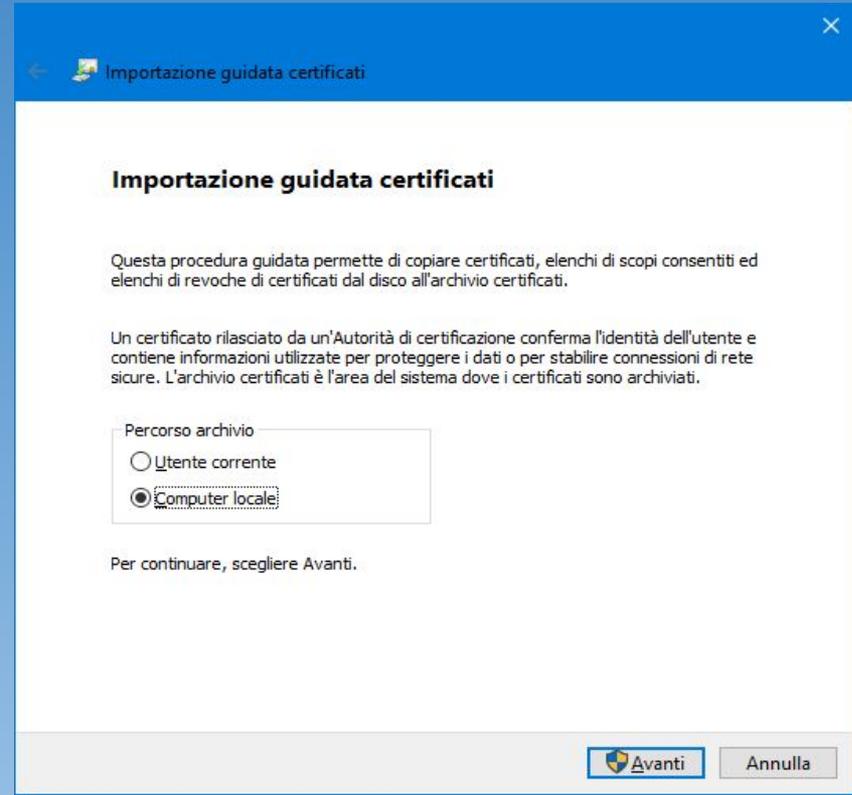
Subject: CN=Ethea S.r.l., C=Italy
Thumbprint: [REDACTED]
Issuer: CN=Ethea S.r.l., C=Italy
Expired Date: 13/02/2021

Clear

Save Cancel Help

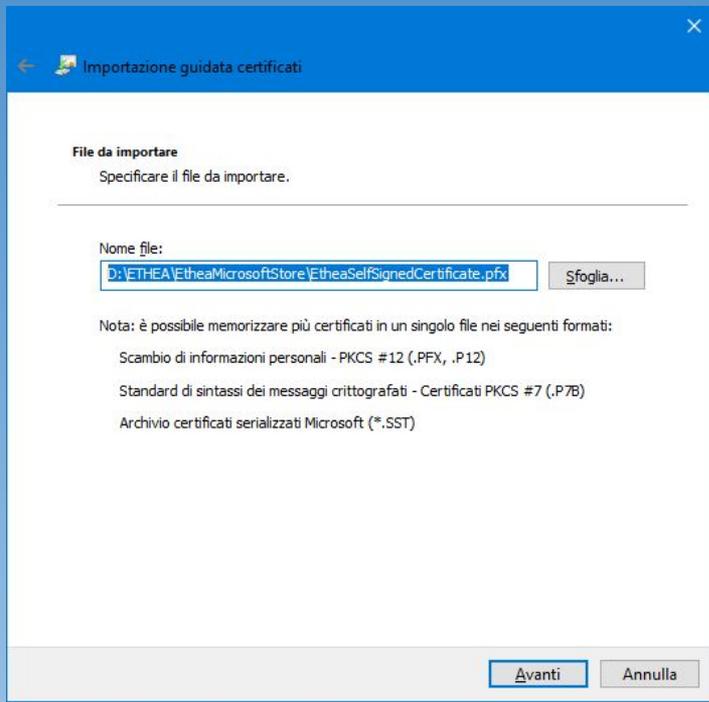
Registrare il certificato

- Per l'installazione sulla propria macchina di sviluppo o dei tester
 - doppio click sul file pfx:
- N.B. Ripetere questo processo anche sulle macchine sulle quali volete distribuire il pacchetto appx.



Registrazione il certificato

- Selezionare file e immettere la password usata per crearlo:



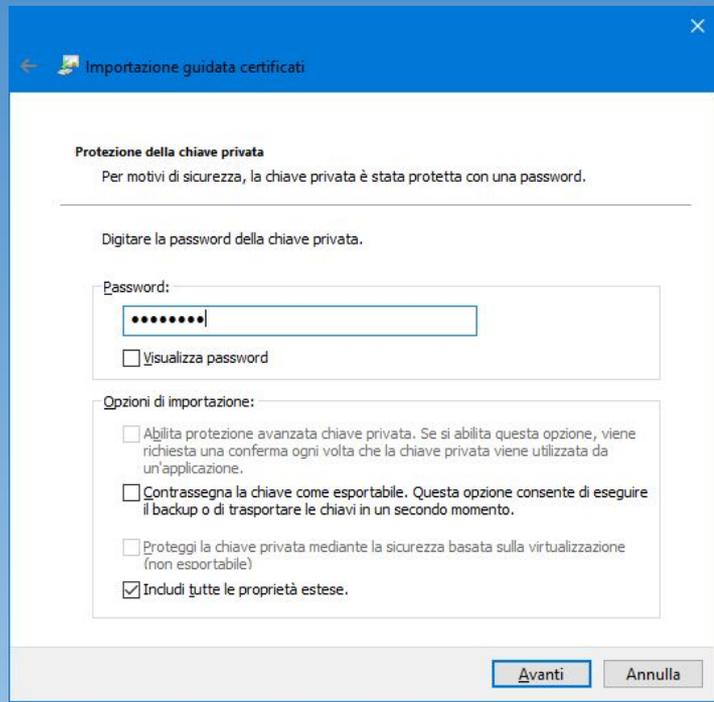
Importazione guidata certificati

File da importare
Specificare il file da importare.

Nome file:

Nota: è possibile memorizzare più certificati in un singolo file nei seguenti formati:

- Scambio di informazioni personali - PKCS #12 (.PFX, .P12)
- Standard di sintassi dei messaggi crittografati - Certificati PKCS #7 (.P7B)
- Archivio certificati serializzati Microsoft (*.SST)



Importazione guidata certificati

Protezione della chiave privata
Per motivi di sicurezza, la chiave privata è stata protetta con una password.

Digitare la password della chiave privata.

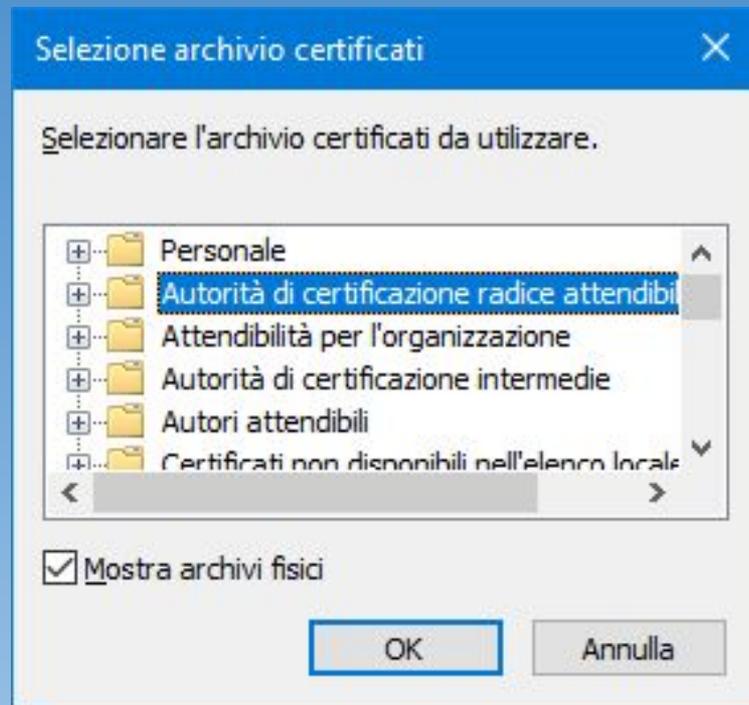
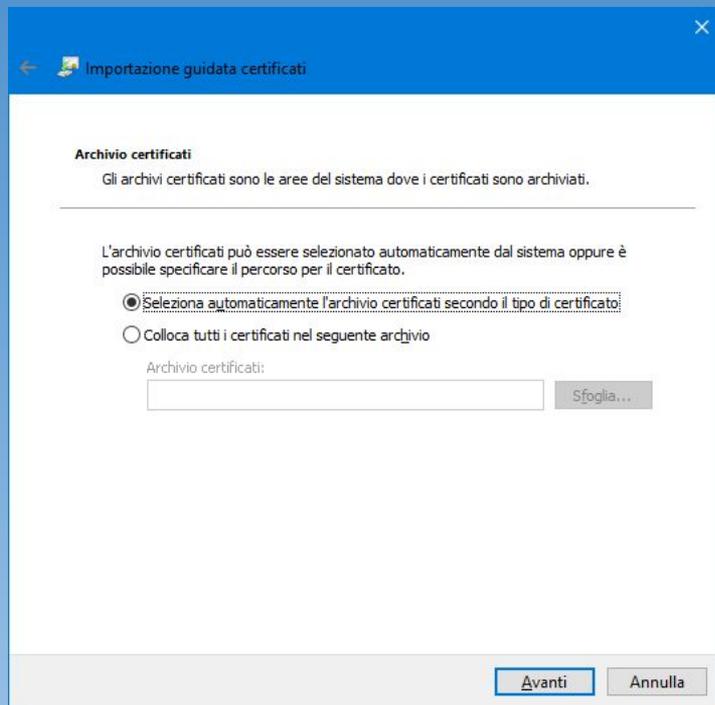
Password:
 Visualizza password

Opzioni di importazione:

- Abilita protezione avanzata chiave privata. Se si abilita questa opzione, viene richiesta una conferma ogni volta che la chiave privata viene utilizzata da un'applicazione.
- Contrassegna la chiave come esportabile. Questa opzione consente di eseguire il backup o di trasportare le chiavi in un secondo momento.
- Proteggi la chiave privata mediante la sicurezza basata sulla virtualizzazione (non esportabile).
- Includi tutte le proprietà estese.

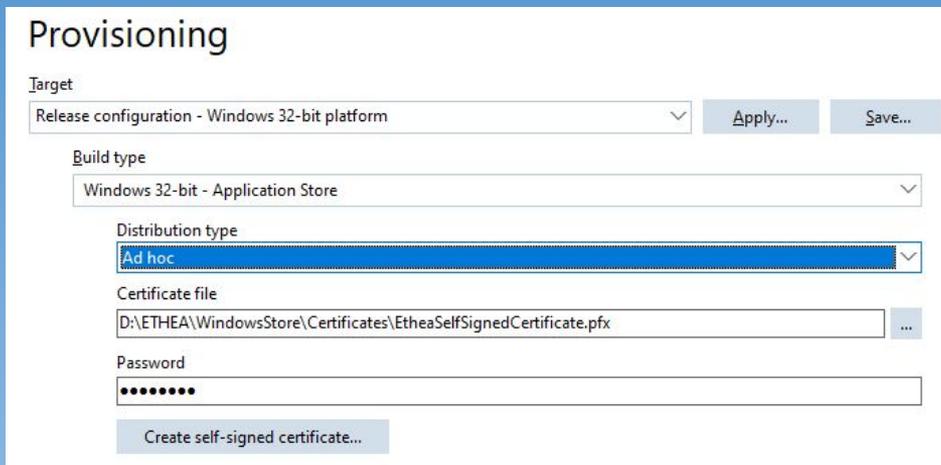
Registrare il certificato

- Inserire il certificato nell'autorità di cert.radice attendibile:



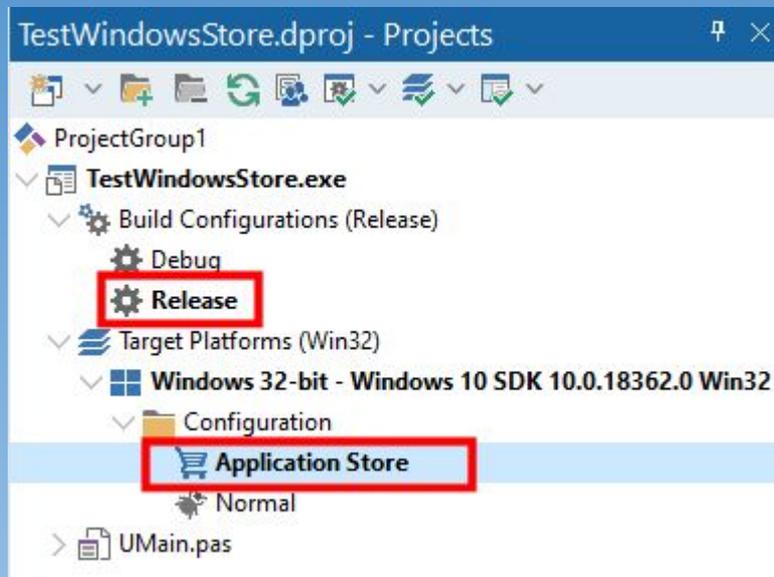
Build options del pacchetto AdHoc

- Project/Options/Provisioning:
 - Ad hoc



The screenshot shows the 'Provisioning' dialog box. The 'Target' dropdown is set to 'Release configuration - Windows 32-bit platform'. The 'Build type' dropdown is set to 'Windows 32-bit - Application Store'. The 'Distribution type' dropdown is set to 'Ad hoc'. The 'Certificate file' field contains the path 'D:\ETHEA\WindowsStore\Certificates\EtheaSelfSignedCertificate.pfx'. The 'Password' field is masked with dots. There is a 'Create self-signed certificate...' button at the bottom.

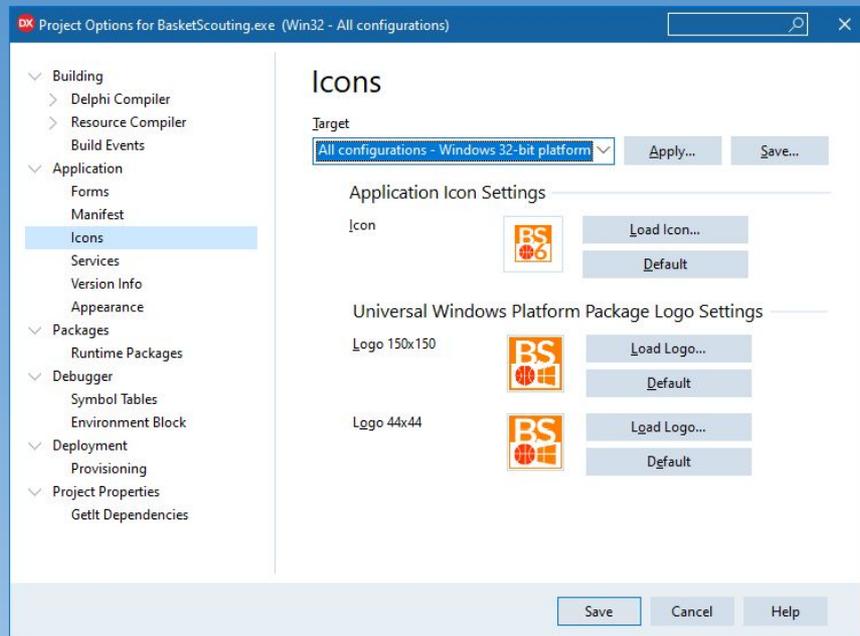
- Build options:



Gestione Icone

- Nell'IDE di Delphi si possono impostare solo:
 - Icona applicazione (usare .ico multires 16,32,44,48,64,128,150,256)
(anche per app Win32 normali)
 - <https://icoconvert.com/>
 - Icone per il Windows Store
 - Logo 150x150
 - Logo 44x44

(queste icone hanno la priorità all'icona dell'applicazione)



Build + Deploy + Packaging + Signing

Build

Project:
D:\ETHEA\BasketScouting6\Projects\D10_3\BasketScouting.dproj

Compiling:
BasketScouting.dpr

Hints: 0 Warnings: 0 Errors: 0

Current Line 47 Total Lines 0

Automatically close on successful compile Cancel

1

Deploy

Project:
D:\ETHEA\BasketScouting6\Projects\D10_3\BasketScouting.dproj

Deploying:
..\..\Templates\USA\HTML\match_summary.css

Hints: 0 Warnings: 0 Errors: 0

Current Line 0 Total Lines 0

Automatically close on successful compile Cancel

2

Deploy

Project:
D:\ETHEA\BasketScouting6\Projects\D10_3\BasketScouting.dproj

Packaging:
..\..\exe\BasketScouting\bin\BasketScouting-unsigned.appx

Hints: 0 Warnings: 0 Errors: 0

Current Line 0 Total Lines 0

Automatically close on successful compile Cancel

3

Deploy

Project:
D:\ETHEA\BasketScouting6\Projects\D10_3\BasketScouting.dproj

Signing:
..\..\exe\BasketScouting\bin\BasketScouting-unsigned.appx

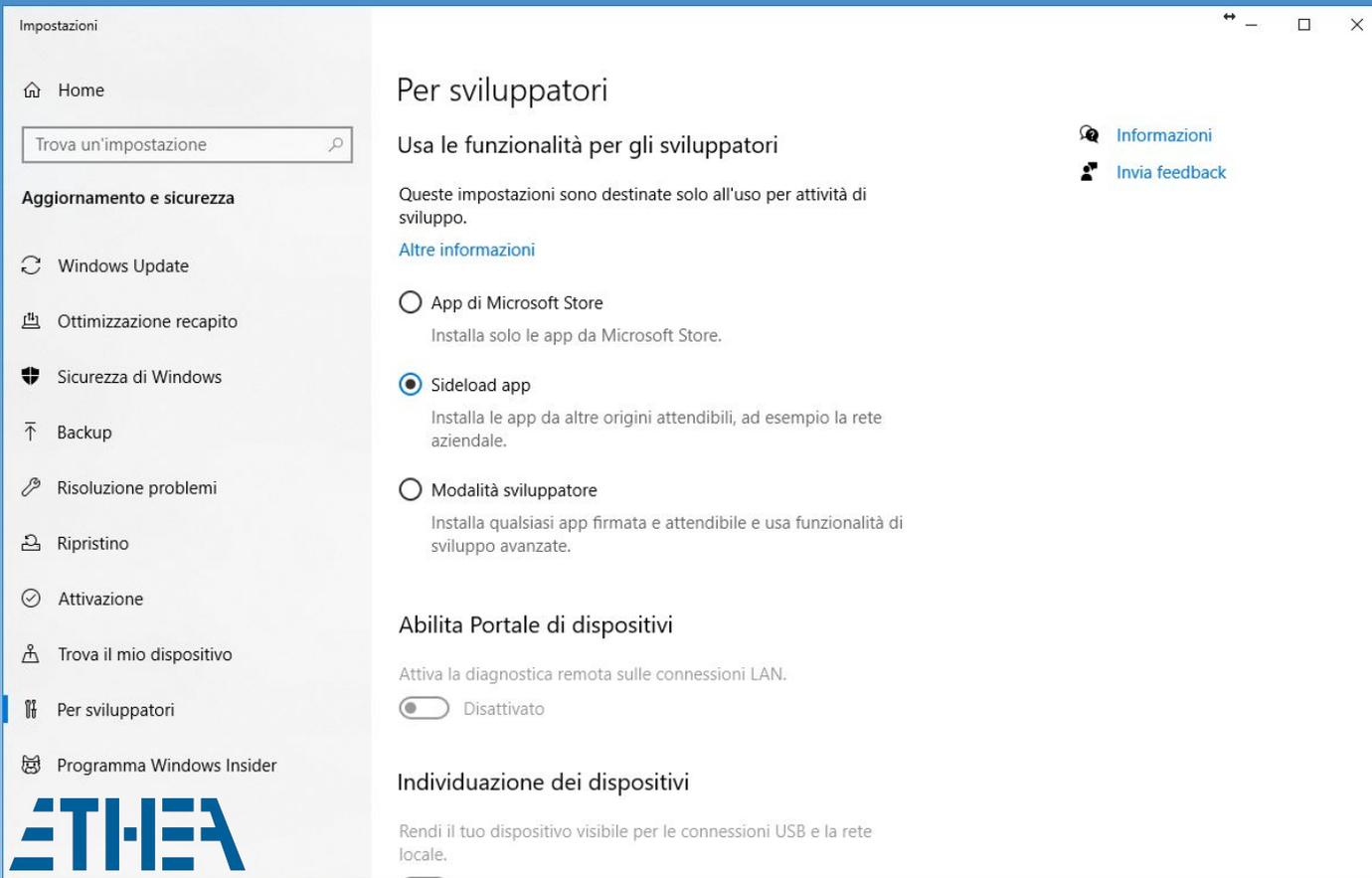
Hints: 0 Warnings: 0 Errors: 0

Current Line 0 Total Lines 0

Automatically close on successful compile Cancel

4

Sblocco installazione Ad Hoc



The screenshot shows the Windows Settings application in Italian, specifically the 'Per sviluppatori' (Developer) section. The left sidebar contains various settings categories, with 'Per sviluppatori' highlighted. The main content area is titled 'Per sviluppatori' and includes a search bar, a description of developer settings, and three radio button options: 'App di Microsoft Store', 'Sideload app' (which is selected), and 'Modalità sviluppatore'. Below these are sections for 'Abilita Portale di dispositivi' and 'Individuazione dei dispositivi'. The 'Sideload app' option is selected, indicating that apps can be installed from other sources like a corporate network.

Impostazioni

Home

Trova un'impostazione

Aggiornamento e sicurezza

- Windows Update
- Ottimizzazione recapito
- Sicurezza di Windows
- Backup
- Risoluzione problemi
- Ripristino
- Attivazione
- Trova il mio dispositivo
- Per sviluppatori**
- Programma Windows Insider

Per sviluppatori

Usa le funzionalità per gli sviluppatori

Queste impostazioni sono destinate solo all'uso per attività di sviluppo.

[Altre informazioni](#)

- App di Microsoft Store
Installa solo le app da Microsoft Store.
- Sideload app
Installa le app da altre origini attendibili, ad esempio la rete aziendale.
- Modalità sviluppatore
Installa qualsiasi app firmata e attendibile e usa funzionalità di sviluppo avanzate.

Abilita Portale di dispositivi

Attiva la diagnostica remota sulle connessioni LAN.

Disattivato

Individuazione dei dispositivi

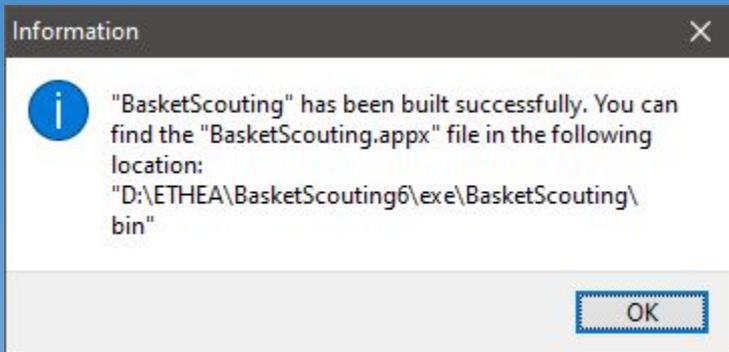
Rendi il tuo dispositivo visibile per le connessioni USB e la rete locale.

[Informazioni](#)

[Invia feedback](#)

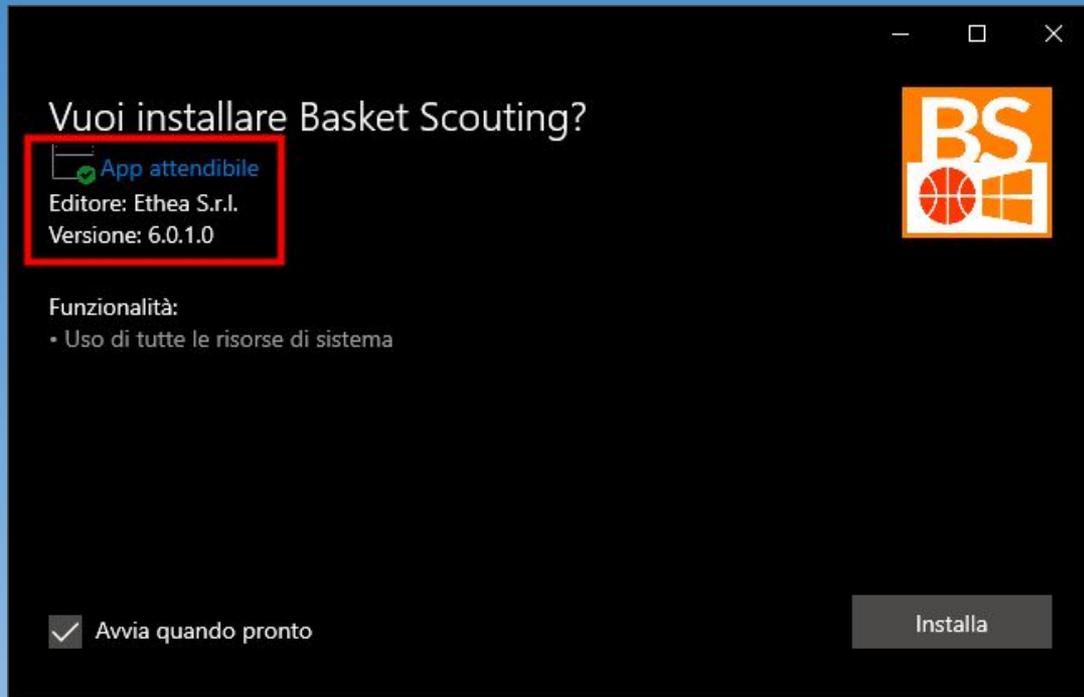
Per installare pacchetti “ad-hoc” dovete consentirlo dal pannello di windows: “Sideload app”

Build + Deploy + Packaging + Signing



Pacchetto pronto per l'installazione sulla propria macchina o sulle macchine di test con il certificato self-signed installato.

Lanciare il file MioProgetto.appx per installarlo sulla macchina.



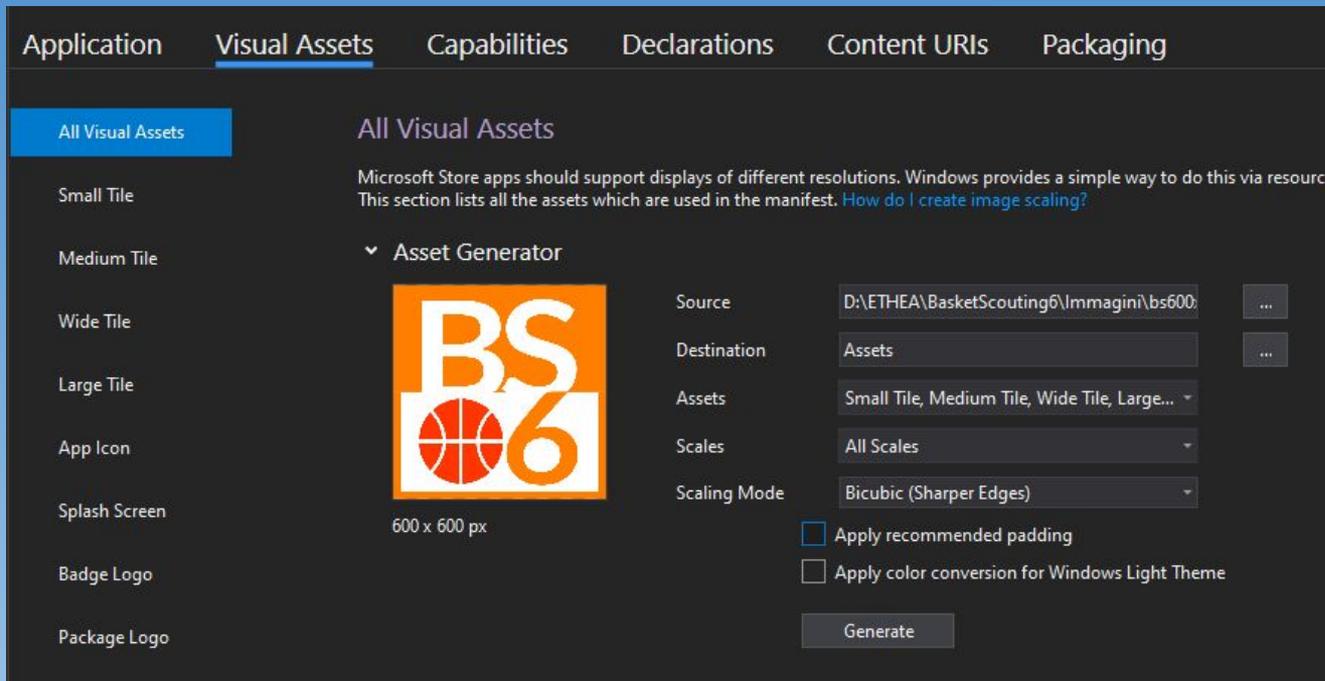
OPERAZIONI AVANZATE

Soluzioni a problemi comuni...

- In questa sezione vedremo:
 - La gestione delle icone di una app UWP
 - Come associare estensioni di file alla propria app
 - Come gestire le DLL ActiveX (DLL Sideloadng)
 - Come gestire i dati utente
 - Come gestire i Font custom
 - Come automatizzare Build/Deploy/Signing

Gestione Icone con Visual Studio

- Nuovo progetto in Visual Studio Community:



- Base: 600x600 px
- Scelta Assets
- Scelta scales
- Algoritmo scaling
- no“reccomended padding”
- Wide Tile
- App Icon
- Store Icon

Custom Appx Manifest

Manifest di default:

- 2 Icone
- versione
- No Publisher
- 1 sola lingua

Usiamo un custom xml manifest per superare i vincoli imposti dall'IDE i Delphi!

Custom Manifest:

- **AppxManifest.template.xml**
 - nella stessa cartella del progetto!
- In Visual Studio: 55 Icone
 - Icona App
 - Icona Store
 - Icone Tiles
- Più lingue
- Associazione estensioni files

Custom Appx Manifest

- Identity e Store logo:

```
<Properties>  
  <DisplayName>Basket Scouting</DisplayName>  
  <PublisherDisplayName>Ethea S.r.l.</PublisherDisplayName>  
  <Description>Scout your basket team</Description>  
  <Logo>Assets\StoreLogo.png</Logo>  
</Properties>
```

- Lingue aggiuntive (impatta sulla gestione nello store):

```
<Resources>  
  <Resource Language="en-US" />  
  <Resource Language="it-IT" />  
</Resources>
```

Appx Manifest: icons name e tiles

- Visual elements:

```
<uap:VisualElements
```

```
  DisplayName="Basket Scouting"
```

```
  Square150x150Logo="Assets\Logo150x150.png"
```

```
  Square44x44Logo="Assets\Logo44x44.png"
```

```
  Description="Scout basket teams"
```

```
  BackgroundColor="transparent">
```

```
  <uap:DefaultTile Square71x71Logo="Assets\SmallTile.png" Square310x310Logo="Assets\LargeTile.png"
Wide310x150Logo="Assets\Wide310x150.png" ShortName="Basket Scouting">
```

```
    <uap:ShowNameOnTiles>
```

```
      <uap:ShowOn Tile="wide310x150Logo"/>
```

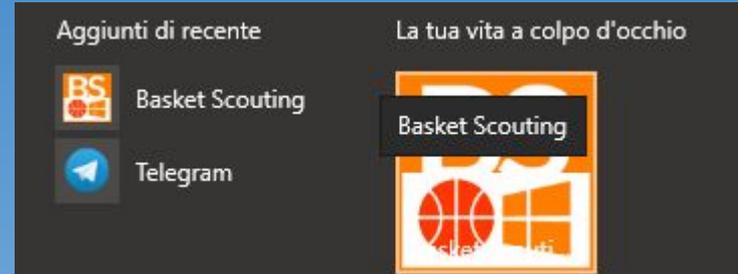
```
      <uap:ShowOn Tile="square310x310Logo"/>
```

```
      <uap:ShowOn Tile="square150x150Logo"/>
```

```
    </uap:ShowNameOnTiles>
```

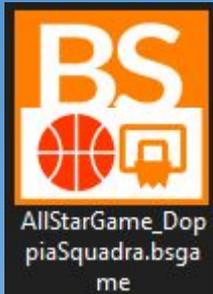
```
  </uap:DefaultTile >
```

```
</uap:VisualElements>
```



Appx Manifest: File associations

- File associations



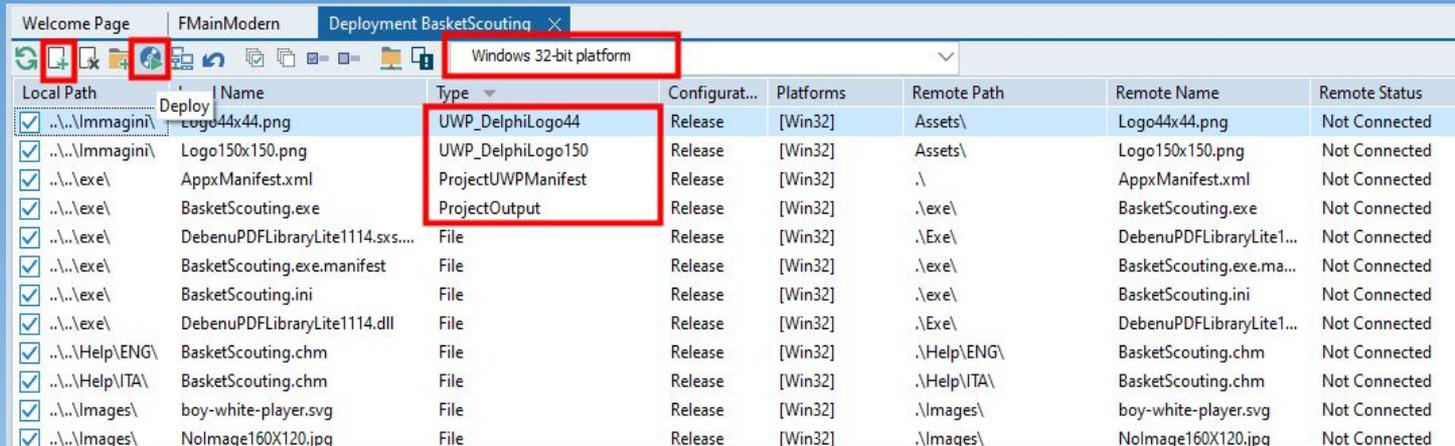
```
<uap:Extension Category="windows.fileTypeAssociation">  
  <uap:FileTypeAssociation Name="basket_scouting_game">  
    <uap:SupportedFileTypes> <uap:FileType>.bsgame</uap:FileType></uap:SupportedFileTypes>  
    <uap:DisplayName>Basket Scouting Game (Display)</uap:DisplayName>  
    <uap:Logo>Assets\bsgame_400.png</uap:Logo>  
    <uap:InfoTip>Basket Scouting Game (Info Tip)</uap:InfoTip>  
  </uap:FileTypeAssociation>  
</uap:Extension>
```



```
<uap:Extension Category="windows.fileTypeAssociation">  
  <uap:FileTypeAssociation Name="basket_scouting_team">  
    <uap:SupportedFileTypes><uap:FileType>.bsteam</uap:FileType></uap:SupportedFileTypes>  
    <uap:DisplayName>Basket Scouting Team (Display)</uap:DisplayName>  
    <uap:Logo>Assets\bsteam_400.png</uap:Logo>  
    <uap:InfoTip>Basket Scouting Team (Info Tip)</uap:InfoTip>  
  </uap:FileTypeAssociation>  
</uap:Extension>
```

Definizione file per il Deploy

- Deploy: Project / Deployment: Selezionare “Windows 32-bit platform”
 - Aggiungere i files (Impostare a mano la “Remote Path” corretta!)
 - Sconsiglio di cambiare il Remote name
 - I tipi diversi da “file” sono gestiti dall’IDE di Delphi



Local Path	Name	Type	Configurat...	Platforms	Remote Path	Remote Name	Remote Status
..\..\Immagini\	Logo44x44.png	UWP_DelphiLogo44	Release	[Win32]	Assets\	Logo44x44.png	Not Connected
..\..\Immagini\	Logo150x150.png	UWP_DelphiLogo150	Release	[Win32]	Assets\	Logo150x150.png	Not Connected
..\..\exe\	AppxManifest.xml	ProjectUWPManifest	Release	[Win32]	.\	AppxManifest.xml	Not Connected
..\..\exe\	BasketScouting.exe	ProjectOutput	Release	[Win32]	.\exe\	BasketScouting.exe	Not Connected
..\..\exe\	DebenuPDFLibraryLite1114.sxs...	File	Release	[Win32]	.\Exe\	DebenuPDFLibraryLite1...	Not Connected
..\..\exe\	BasketScouting.exe.manifest	File	Release	[Win32]	.\exe\	BasketScouting.exe.ma...	Not Connected
..\..\exe\	BasketScouting.ini	File	Release	[Win32]	.\exe\	BasketScouting.ini	Not Connected
..\..\exe\	DebenuPDFLibraryLite1114.dll	File	Release	[Win32]	.\Exe\	DebenuPDFLibraryLite1...	Not Connected
..\..\Help\ENG\	BasketScouting.chm	File	Release	[Win32]	.\Help\ENG\	BasketScouting.chm	Not Connected
..\..\Help\ITA\	BasketScouting.chm	File	Release	[Win32]	.\Help\ITA\	BasketScouting.chm	Not Connected
..\..\Images\	boy-white-player.svg	File	Release	[Win32]	.\Images\	boy-white-player.svg	Not Connected
..\..\Images\	NoImage160X120.jpg	File	Release	[Win32]	.\Images\	NoImage160X120.jpg	Not Connected

Le DLL ActiveX (Sideloading DLL)

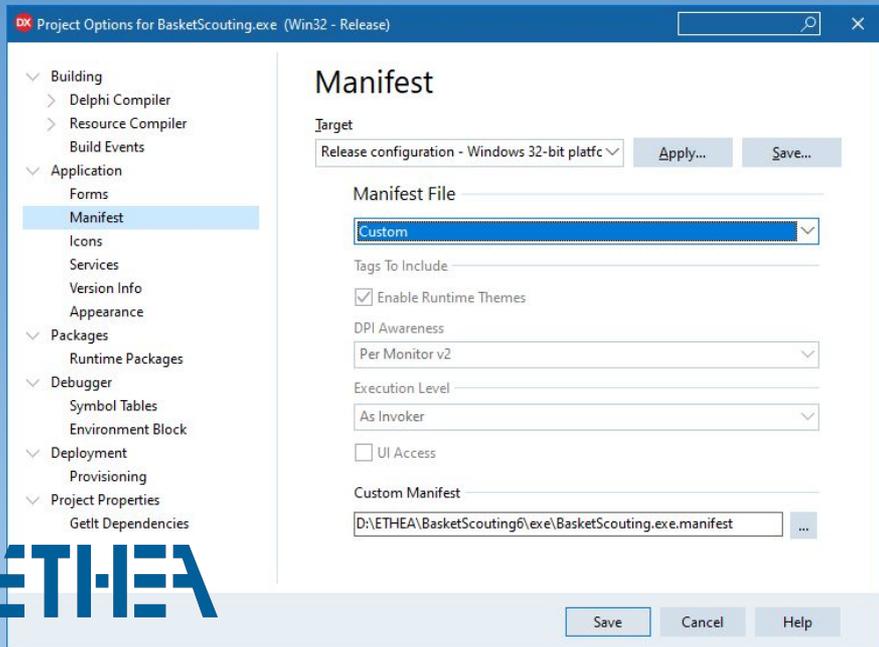
Tecnica usabile a prescindere dalla creazione di una app per lo store, per usare Activex/dll senza doverli registrare.

Project Options:

Application/Manifest: **Custom**

Usare un manifest custom con:

```
<dependency>
  <dependentAssembly>
    <assemblyIdentity
      type="win32"
      name="DebenuPDFLibraryLite1114.sxs"
      version="1.0.0.0" />
    </dependentAssembly>
  </dependency>
```

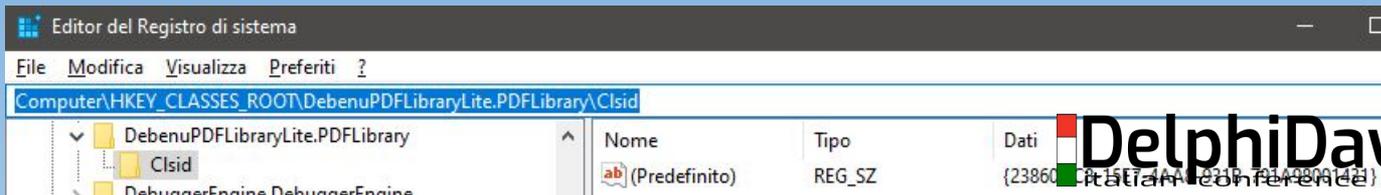


DLL ActiveX (Sideloading DLL)

Manifest legato all'ActiveX/dll:

- dll nella cartella dell'exe (es.DebenuPDFLibraryLite1114.dll)
- Nome del file: xxx.sxs.manifest (DebenuPDFLibraryLite1114.sxs.manifest)
- Il clsid della dll si trova nel registry:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">  
<assemblyIdentity type="win32" name="DebenuPDFLibraryLite1114.sxs" version="1.0.0.0" />  
<file name="DebenuPDFLibraryLite1114.dll">  
  <comClass description="DebenuPDFLibraryLite1114.PDFLibrary Object"  
    clsid="{23860CC8-15E7-4AA8-931B-791A98001431}" threadingModel="Apartment" />  
</file>  
</assembly>
```



Installazione di Font aggiuntivi

Utilizzando **IconFontImageList** per le icone e un font specifico per il tabellone delle partite occorre fornire i font “custom” all’avvio dell’applicazione:

```
procedure AddFont(const AFontName: string);
var LFontFileName: string;
begin
  LFontFileName := Config.ImagesPathDelim+AFontName;
  if FileExists(LFontFileName) then
  begin
    AddFontResource(PWideChar(LFontFileName));
    SendMessage(HWND_BROADCAST, WM_FONTCHANGE, 0, 0);
  end
  else
    raise Exception.CreateFmt(FONT_NOT_FOUND, [LFontFileName]);
end;
```



La gestione dei dati Utente

Cartella di installazione dell'applicazione:

- `C:\Program Files\WindowsApps\BasketScouting_6.0.0.0_x86__jw4rrkswmx8cc`
- Dati locali (consigliato cartella Roaming)
- Dati temporanei (cartella Temp)
- La App ha i privilegi di accesso a tutte le risorse del sistema ma non può mai elevarsi al livello Admin (attenzione a come scrivete la App).
- Accessibilità cartella dati utente: `{userdoc}\MiaApp\Samples`
 - Non si può definire nel deploy se alcuni files debbano finire “altrove”
 - All'avvio, creare la cartella e unzipparci un pacchetto dati “sample”

Automatizzare Build/Deploy/Signing

- Limiti approccio automatico:
 - Operazione lenta (via paclient...)
 - Gestione file .deployproj in source-ctrl complesso (i file si spostano!)
 - **Limite di numero di files** (processo di Packaging si inchioda!)
- Gestire un proprio file di mapping
 - Due diversi (uno per AdHoc e uno per WinStore: cambia il manifest)
- Definire script di Build/Packaging/Signing
 - Packaging/Deploy AdHoc (sintassi):

```
"makeappx.exe" pack -f "MapFile_AdHoc.tmp" -p "unsigned.appx"
```

```
"signtool.exe" sign -f "SelfSignedCertificate.pfx" -p "password" -fd SHA256 "unsigned.appx"
```

- Il pacchetto per lo Store non deve essere firmato!

CONFIGURAZIONE DELLO STORE

Come interagire con lo Store

- **Nello Store occorre definire:**
 - Riservare un nome per la propria App (già fatto)
 - Creare il primo “Invio”
 - Prezzi e disponibilità
 - Proprietà (categoria e requisiti di sistema)
 - Classificazione e fasce di età
 - Pacchetti (dove caricare il proprio: check dati del publisher)
 - Presentazioni nello store (screenshot, icone, banner, trailer...)
 - Gestione multilingua (se è stata configurata nel manifest)
 - Gestione “add-on” (moduli aggiuntivi)

Config.Store: Gestione della app

Maschera di riepilogo:

 **Basket Scouting**
Applicazione

Informazioni sull'applicazione

Componenti aggiuntivi

- Analisi
- Gestione prodotto
- Servizi

Informazioni sull'applicazione

Invii

Invio	Stato	Ultima modifica (UTC)	Azione
Invio 1	Certificazione in corso Visualizza stato	16/05/2020	Annulla

Componenti aggiuntivi

[Crea un nuovo add-on](#)

ID prodotto	Tipo di prodotto	Stato	Ultima modifica (UTC)
DoubleTeam	Bene durevole	Non inviato	29/03/2020

Gestione degli “invii” allo store e dei componenti aggiuntivi

Config.Store: Gestione dell'invio

BS **Basket Scouting**
Applicazione

Informazioni sull'applicazione

Invii ^

Invio 1

Prezzi e disponibilità

Proprietà

Classificazioni per fascia d'età

Pacchetti

Presentazioni nello Store v

Opzioni di invio

Componenti aggiuntivi

Analisi v

Gestione prodotto v

Servizi v

Invio 1

Questo invio è in fase di certificazione. [Visualizza avanzamento](#)

Prezzi e disponibilità
Gratis e disponibile per i clienti. Sola lettura

Proprietà
Sport Sola lettura

Classificazioni per fascia d'età Sola lettura

Pacchetti Sola lettura

BasketScouting.appx Convalidato

Presentazioni nello Store Aggiungi/rimuovi lingue Esporta presentazioni Sola lettura

Lingue supportate nei pacchetti

Inglese (Stati Uniti)	Completo
Italiano (Italia)	Completo

Lingue presentazioni nello Store aggiuntive

Italiano	Completo
----------	----------

Opzioni di invio Sola lettura

Pubblica appena possibile

Config.Store: Gestione degli add-on

Maschera di riepilogo di un add-on (a pagamento):

 **DoubleTeam**
Bene durevole

↔

Informazioni sull'applicazione

Componenti aggiuntivi
DoubleTeam

Invii ^

Invio 1

Prezzi e disponibilità

Proprietà

Classificazioni per fascia d'età

Presentazioni nello Store

Opzioni di invio

Analisi v

Invio 1

Elimina

Proprietà
Bene durevole Completo ✓

Prezzi e disponibilità
9,99 EUR e disponibile per i clienti. Completato ✓

Classificazioni per fascia d'età Completa ✓

Presentazioni nello Store ⓘ [Aggiungi/rimuovi lingue](#) [Esporta presentazioni](#) Completo ✓

Importa presentazioni

Inglese (Stati Uniti) Completo
Italiano Completo

Opzioni di invio Completo ✓

Publica manualmente

Visibilità nello Store

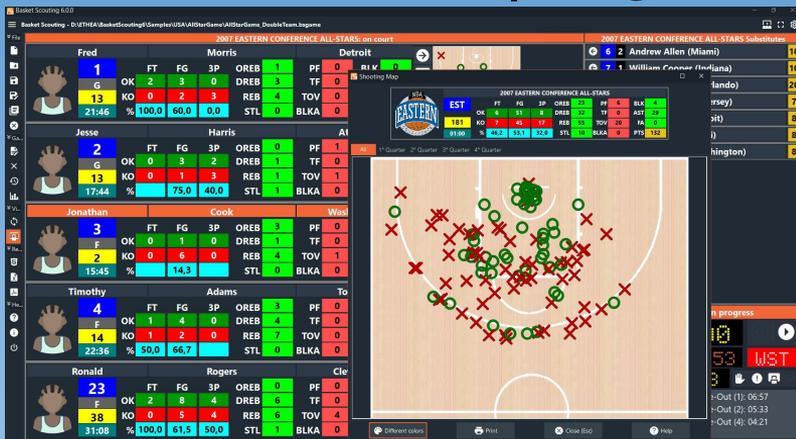
- Link nel browser: <https://www.microsoft.com/it-it/p/basket-scouting/9nqh61r6k0t2>
- Link per accesso nello store: <ms-windows-store://pdp/?productid=9NQH61R6K0T2>

Scaricatela, la versione “base” è gratuita... e soprattutto votate con 5 ☆☆☆☆☆ di gradimento 😊



Una app-moderna e intuitiva...

- Spesso il “vestito” fa la differenza, specialmente se i potenziali clienti decidono sulla base degli screen-shot dell’applicazione e giudicano la app prima di provarla.
- Design moderno, icone “material”, tema chiaro/scuro, High-DPI, multilingua, interfaccia intuitiva, help integrato e ...prezzo basso.





THANK YOU