

Apply a Screensaver Template for Windows 98

Zuhor H. Hasan

Ahmed S. Nori

Asmaa Y. Hamo

ahmed.s.nori

asmahammo

@uomosul.edu.iq

@uomosul.edu.iq

College of Administration
and Economics
University of Mosul

College of Computer Science and Mathematics
University of Mosul, Iraq

Received on: 16/02/2002

Accepted on: 12/10/2002

ABSTRACT

This paper involves designing graphics model for displaying and working under Windows98 operating system called Screen Saver, which is considered as one of the most significant desktop settings. Various and multiple algorithms were used and implemented for designing Screen Saver, with the capabilities of (storing its settings within the Windows98 Registry and preventing using the computer from an unauthorized user).

To perform this work, Visual C++ v6.0 and Windows98 API functions were considered as programming tools. Software was tested successfully under Windows98 operating system.

Keywords: Screen Saver, Registry, API.

تطبيق قالب حافظه شاشة لنظام التشغيل ويندوز 98

أسماء ياسين حمو

أحمد سامي نوري

زهور حسين حسن

كلية علوم الحاسبات والرياضيات

كلية الإدارة والاقتصاد

جامعة الموصل

جامعة الموصل

الملخص

يتضمن البحث تصميم قالب رسومي للعرض يعمل كحافضة شاشة باعتبارها أهم إعدادات سطح المكتب. ومن خصائص هذا القالب الرسومي هو إتاحة الفرصة للمستخدم لاختيار الصور التي يرغب في عرضها لكي تعطي الطابع الشخصي للحاسبة إذ تم تصميم خوارزميات عرض رسومية متعددة مع إمكانية حفظ الإعدادات الخاصة بالحافضة في التسجيل لويندوز 98 ليتسنى استخدامها عند كل عملية تشغيل.

كما يستخدم القالب في حماية الحاسبة من الأشخاص غير المخولين وذلك عن طريق استخدام كلمة المرور في كلا الحالتين (عند بداية التشغيل أو عند ترك الحاسبة وهي تعمل لمدة من الزمن) دون استجابة من المستخدم أي (الضغط على لوحة المفاتيح أو جهاز الفأرة). استخدمت لغة V6.0 ++ Visual C في البرمجة مع الاعتماد على أسلوب واجهة البرمجة التطبيقية API وتعمل بنجاح في بيئة نظام التشغيل ويندوز 98. الكلمات المفتاحية: حافظه الشاشة، تسجيل، API.

المقدمة

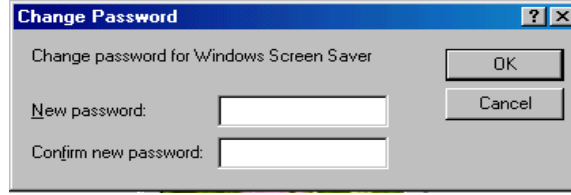
يرتبط نظام التشغيل ويندوز 98 مع المستخدم بواجهة رسومية تدعى واجهة المستخدم الرسومية Graphic User Interface GUI باستخدام أسلوب سطح المكتب إذ يتفاعل المستخدم مع النظام من خلال كائنات ورموز متنوعة [5].

ويتم إعداد سطح المكتب لتأخذ الحاسبة الطابع الشخصي لمستخدمها وأحد أهم هذه الإعدادات هو حافظه الشاشة . ومبدأ عمل حافظه الشاشة هو انه خلال فترة زمنية سبق أن حددها المستخدم لم يحدث فيها أي استعمال لجهاز الفأرة أو للوحة المفاتيح فيجرب تفعيل برنامج حافظه الشاشة أوتوماتيكياً الذي يفترض أن يعرض رسومات متحركة وبألوان مختلفة وذلك لمنع اشتعال الفسفور من جراء الشاشة المتوقفة عن العمل [8] . ولغرض إعادة محتويات الشاشة الأصلية إلى وضعها يمكن الضغط على أي مفتاح في لوحة المفاتيح أو تحريك جهاز الفأرة.

كما يستخدم برنامج حافظه الشاشة لحماية الحاسبة من التطفل عند تركها لفترة من الزمن وذلك بخزن مسبق لكلمة المرور Password . بعد تشغيل برنامج حافظه الشاشة فان أي استخدام للمفاتيح أو تحريك لجهاز الفأرة لن يعيد محتويات الشاشة الأصلية إلا بعد إعطاء كلمة المرور الصحيحة [9].

يحتوي نظام التشغيل ويندوز 98 على عدد من حافظات الشاشة وجميعها تستخدم ملفات صورية نوع BMP و PCX. محددة مسبقاً من قبل النظام و لا تعطي أي طابع شخصي للمستخدم إلا إذا استخدم الحافظة الخاصة بالنص ثلاثي الأبعاد [6].

هذا من جهة ومن جهة أخرى فان نظام التشغيل ويندوز 98 يفتقر إلى الحماية الضرورية من المتطفلين وخاصة عند بدء التشغيل. إما كلمات المرور الموجودة في لوحة التحكم فلأغراض تسجيل الدخول فقط ويمكن تجاوزها بالضغط على المفتاح الموجود في مربع حوار كلمات المرور كما في الشكل (1) [9].

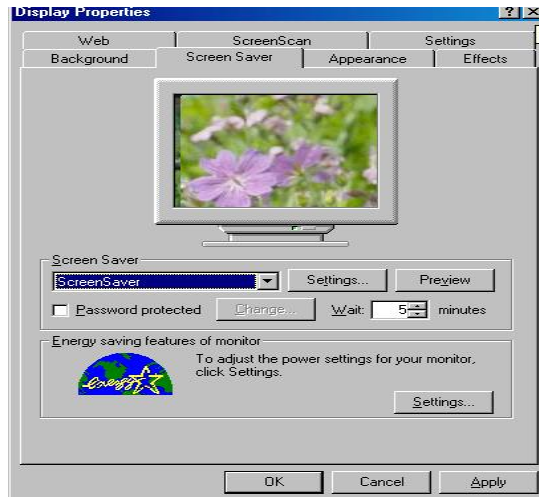


الشكل (1): مربع حوار كلمات السر

ولإضفاء الطابع الشخصي على الحاسبة لابد من تصميم قالب يكون بمثابة حافظه شاشة يمكن المستخدم من اختيار الصور التي ينوي عرضها. ولما كانت هذه الصور ثابتة مع العلم أن أساس عمل حافظه الشاشة هو عرض الرسوم المتحركة وبألوان مختلفة لذلك تحتم اختيار خوارزميات خاصة لعرض الصور لتبدو متحركة. كما يمكن استخدام هذه الحافظه بشكل معين بحيث تمنع الدخول إلى الحاسبة واستعمالها إلا من قبل المخولين أي بعد إعطاء كلمة السر المناسبة .

تثبيت برنامج SCR .Screen saver من قبل المستخدم وطريقة إعداده

لكي يعمل التطبيق بشكله الصحيح، لابد من وضع الملف (Screensaver) في الدليل C:\Windows\System . تتم عملية تشغيله عن طريق الوصول إلى خصائص الشاشة ضمن لوحة التحكم ومن ثم اختيار قائمة screensaver التي تظهر عدداً من حافظات الشاشة التي تعمل مع ويندوز ومن ضمنها سيظهر اسم البرنامج مع مجموعة الاختيارات كما موجود في الشكل (2).



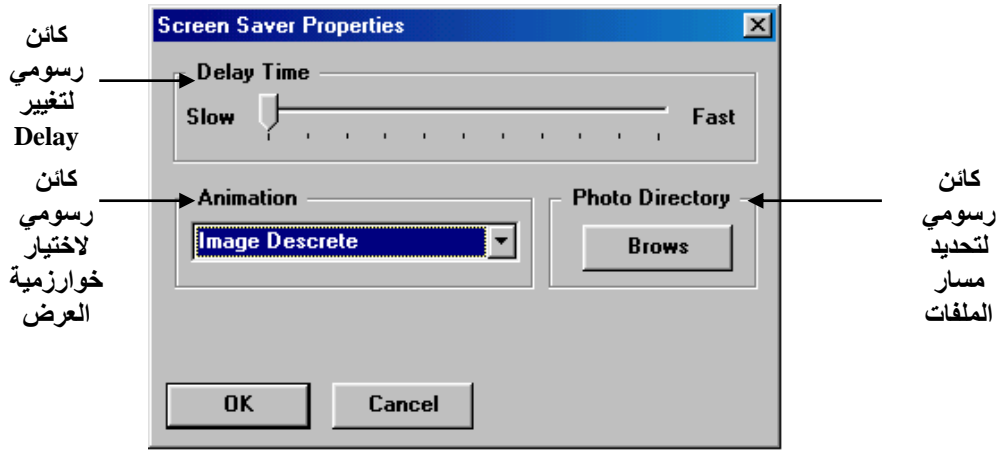
الشكل (2): ظهور التطبيق ضمن اختيارات حافظه الشاشة

عند الضغط على الإعدادات Settings سيظهر مربع حوار التطبيق، الشكل (3) ويشمل ما يأتي:

Delay Time : يستخدم لتحديد سرعة العرض من خلال شريط يمكن تمريره عبر تدرجات مؤشرة، أما slow لتبطئ من سرعة العرض أو Fast لتسريع العرض.

ب- **Animation** : لتحديد خوارزمية العرض إذ تم توفير خمس خوارزميات للعرض وهي: Two ، Two Halfs ، Four Sequres Input، Left to Right ، Image Discrete .Halfs Discrete

ج- **Photo Directory** : قد يرغب المستخدم في عرض أية صورة والتي قد لا تتوفر ضمن الدليل Windows ، عندئذٍ تتم الاستعانة بالأمر Brows لغرض تحديد مسار الملف المطلوب [7].

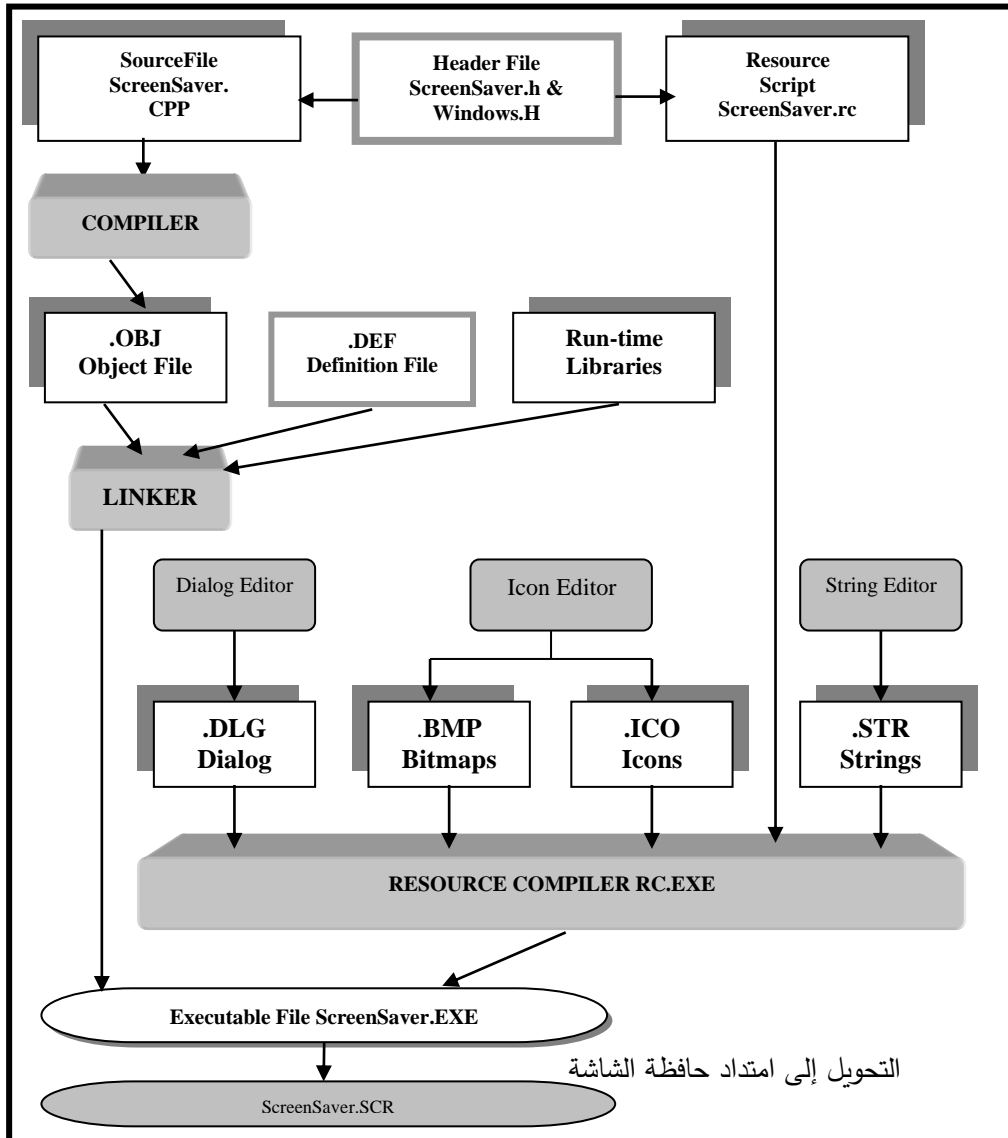


الشكل (3): صندوق حوار التطبيق

أما عند الضغط على المربع الصغير الموجود بجانب Password protected ، الشكل (2) فسوف يظهر مربع الحوار الذي يطلب إدخال كلمة المرور كما في الشكل (1). إذا رغبت المستخدم في استعمال الحافظة أسلوباً لمنع دخول غير المخولين فيتم ذلك بوضع اسم الملف Screensaver.SCR (بعد تثبيت كلمة المرور) بالدليل C:\Windows\Allusers\StartMenu\Programs\Startup. ففي هذه الحالة وعند بداية التشغيل وبعد أن يكتمل تحميل النظام سوف يعمل برنامج حافظة الشاشة وعندها لن يتمكن المستخدم من استعمال الحاسبة إلا بعد إعطاء كلمة السر الصحيحة التي سبق خزنها في الإعدادات الموضحة أعلاه.

ملفات التطبيق

استخدمت لغة Visual C++ ذات الإصدار 6.0 مع دوال واجهة البرمجة التطبيقية الـ API ذات البيئة 32-bit ، وهذه اللغة تحتاج إلى ثلاثة أنواع من الملفات لتكوين الملف النهائي التشغيلي، وهي ملف (المستلزمات) Resource File، ملف (التمهيدات) Header File، ملف (البرنامج) Source File. والشكل (4) يبين جميع أجزاء التطبيق من مرحلة الإنشاء ولغاية الحصول على الملف التنفيذي [1][3].



الشكل (4) : تفاعل مكونات التطبيق مع بعضها للحصول على الملف التطبيقي

(أ) ملف المستلزمات (Resource File)

يصف هذا الملف أنواعاً مختلفة من الكائنات التي تستخدم في ويندوز والتي توضع في الذاكرة وحسب الحاجة إليها، وهذه الأجزاء أو الكائنات يوظفها المستخدم للتفاعل مع ويندوز . لقد تم استخدام أنواع من هذه الكائنات أو المصادر لإنشاء هذا التطبيق والموضحة في الشكل(3).

أولاً: أيقون التطبيق Icon .

ثانياً: سلسلة حرفية String .

ثالثاً: صندوق الحوار Dialog Box .

تمثل المتغيرات الموجودة داخل صندوق الحوار (إعدادات التطبيق) والتي على أساسها يكون التشغيل بحيث يتم إعدادها من قبل المستخدم وبما يمثله كل متغير فيها وكالاتي:

متغير يحدد الفترة الزمنية لتشغيل رسالة الزمن Delay Time .

متغير يحدد نوع خوارزمية العرض Animation .

متغير لتحديد الدليل الخاص بالصور النقطية المطلوب عرضها Photo Directory .

رابعاً: الصور النقطية Bitmaps .

تم استخدام مجموعة من الصور النقطية لغرض العرض، ليس عن طريق ربطها بملف المستلزمات وذلك للحفاظ على التطبيق من ثبات الصور فيه وللتقليل من حجمه، بل يكون ربطها بالبرنامج مباشرة في حالة التنفيذ إذ يتم تحديد المسار لدليل معين من خلال صندوق الحوار وبعدها تجمع الصور ذات الامتداد BMP و PCX. وعلى هذا الأساس يتم عرضها في التطبيق [10].

(ب) ملف التمهيدات Header File (ScreenSaver.H)

يحتوي الأسماء التعريفية لكل الكائنات المستخدمة داخل ملف المستلزمات، إذ أن كل كائن معرف داخل ملف المستلزمات تعطى له قيمة لكي يتعامل معه البرنامج عن طريق هذا التعريف، أي أن جميع الرسومات الموجودة ضمن صندوق حوار البرنامج قد تم تعريفها في هذا الملف، كما انه يعرف في بداية البرنامج كي يتمكن التطبيق من استخدام المصادر [4].

(ج) ملف البرنامج الرئيس (ScreenSaver.CPP)

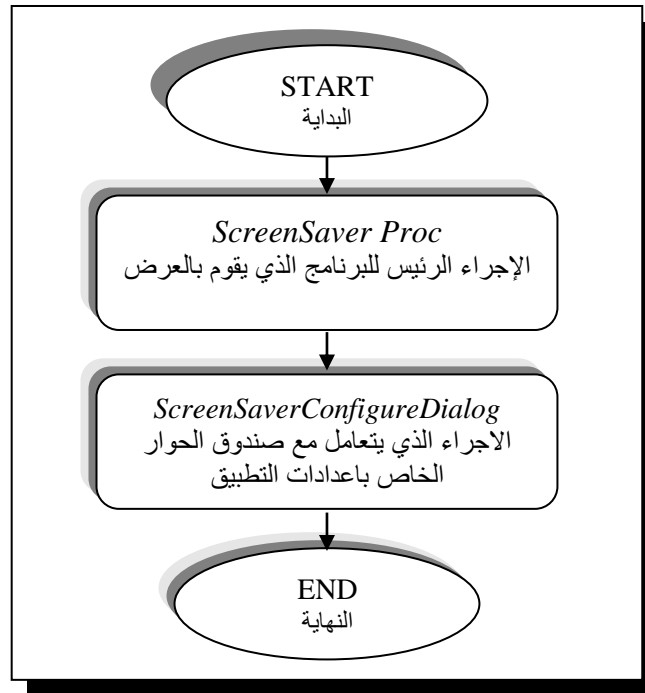
يعد هذا الملف برنامجاً رئيساً للتطبيق والذي كتب بلغة ++ VC ، ويتضمن مجموعة الاستدعاءات لدوال واجهة البرنامج التطبيقية الـ API لبيئة 32-bit، التي تضمنت دوال واجهة الجهاز الرسومية GDI لغرض العرض على الشاشة علاوة على وجود دوال أخرى لـ API لغرض التعامل مع تسجيل ويندوز 98 التي من خلالها تم إنشاء قاعدة بيانات خاصة بالتطبيق لحفظ الإعدادات فيها [2].

إجراءات البرنامج الرئيسية

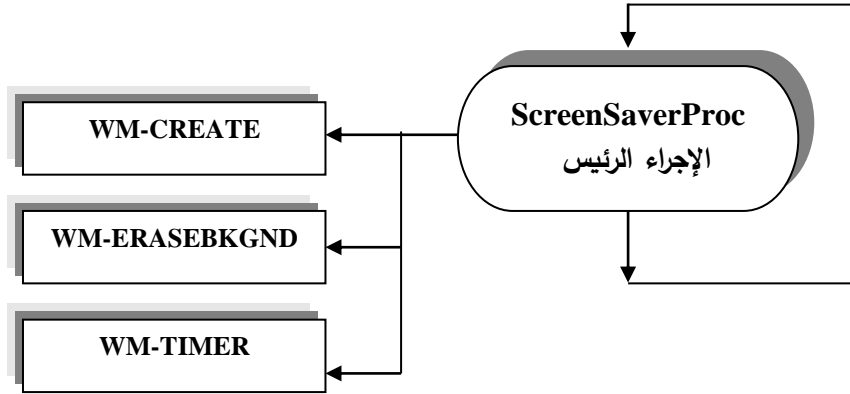
يتألف البرنامج من إجرائين هما : لاحظ الشكل (5)

(أ) إجراء حافظه الشاشة Screen Saver Proc

يعد هذا الإجراء القسم التنفيذي للبرنامج ويعتمد على دالة البرنامج الرئيسية () Screen SaverProc. إذ يقوم باستقبال الرسائل وإرسالها إلى نافذة تطبيق حافظه الشاشة المحددة ، يتكون هذا الإجراء من عدد من الرسائل وهي: الشكل (6).



الشكل (5) : أقسام البرنامج الرئيسية



الشكل (6) : أهم الرسائل في الإجراء الرئيس

أولاً: رسالة التكوين WM-CREATE

ترسل هذه الرسالة في حالة استجابة التطبيق لتكوين نافذته إذ يستجيب الإجراء لهذه الدالة ولكن قبل أن تصبح النافذة مرئية . ومن خلال هذه الرسالة تم تكوين المتغيرات الرئيسة والضرورية للمعالجة فضلاً عن :

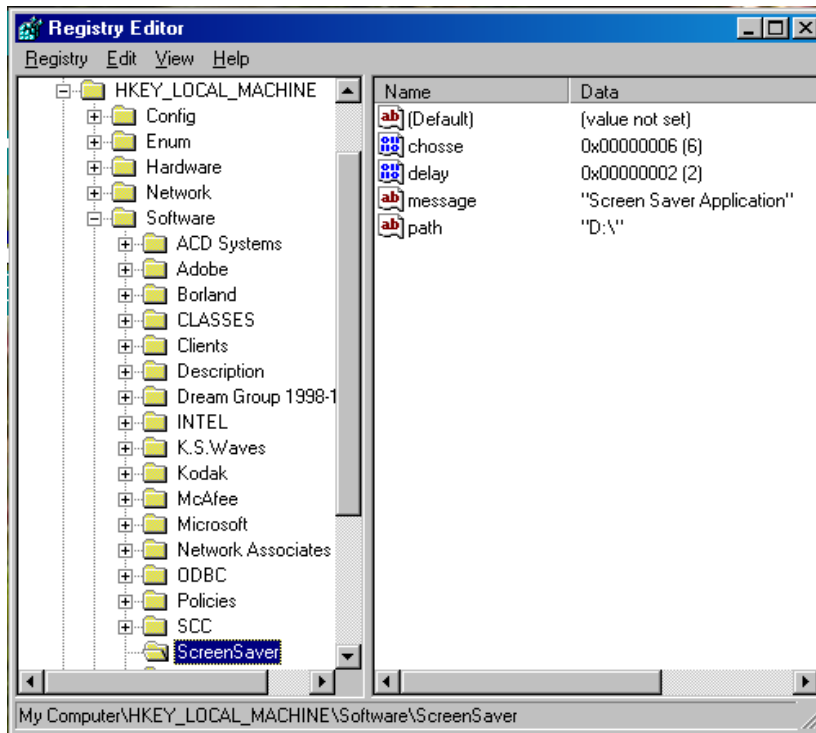
- تكوين قاعدة بيانات التطبيق في التسجيل تحت المفتاح الفرعي Screensaver والمندرج من المفتاح الفرعي Software والتابع للمفتاح الجذري -HKEY- LOCAL_MACHINE ، إذ أن إعدادات التطبيق تؤخذ من هذا المفتاح والشكل (7) يبين بيانات هذا المفتاح من خلال REGEDIT .
- العمل على تشغيل المؤقت الزمني لفترة تأخير محددة والذي عن طريقه تتم الاستجابة للرسالة الخاصة بالمؤقت ومعالجة الصورة.
- عملية تحميل للكائن الرسومي "فرشاة الرسم" وبلون معين .

ثانياً: رسالة مسح الخلفية WM-ERASEBKGND

من خلال هذه الرسالة تمت عملية المسح لخلفية النافذة باستخدام فرشاة الرسم المحملة من قبل الرسالة السابقة ، إذ تم استخدام محتوى الجهاز Device Context DC لتحميل النافذة قبل عملية الرسم وتم الغاء محتوى الجهاز المحمل بعد الانتهاء من عملية الرسم لخلفية النافذة .

ثالثاً: رسالة المؤقت او الزمن WM-TIMER

المعالجة لهذه الرسالة وعملية الاستجابة لها هي جوهر البرنامج وتتضمن عملية العرض الرسومي الصوري بطرق متنوعة. هذه الرسالة يتم ارسالها الى التطبيق خلال فترة معينة محددة من قبل المستخدم في قاعدة البيانات بعد وضعها في دالة تشغيل الزمن ، وهذه الدالة بدورها تعمل على ارسال هذه الرسالة حسب الفترة المستخدمة فيها وتكون في المتغير Delay ، أي ان كل فترة يتم ارسالها تعني عرضاً لصورة جديدة بطريقة مختلفة . ويعزى استخدام ويندوز لهذه الرسالة الى امكانية تثبيت الوقت الذي عنده سيقطع البرنامج .



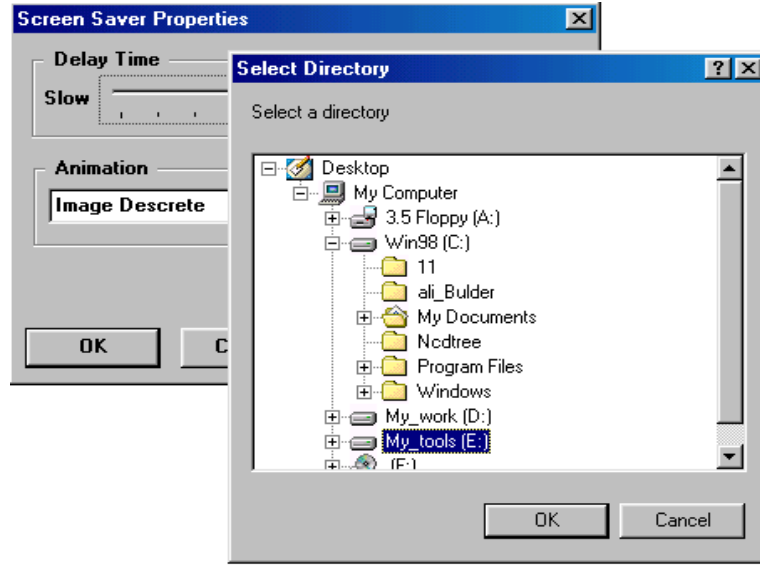
الشكل (7) : بيانات المفتاح screensaver داخل التسجيل

هناك متطلبات يجب توفيرها أو إجراؤها لكي تتم المعالجة الصورية وهي:

- تحميل محتوى جهاز للذاكرة .
- تعريف هيكل للصورة النقطية لاخذ معلومات الصورة منها .
- تحميل الصورة في الهيكل المعرف .

- تحديد او الحصول على الكائن الرسومي المستخدم .
- تكوين محتوى الجهاز المتوافق .
- اختيار الكائن المستخدم ووضعه في محتوى الجهاز DC .
- تحديد خوارزمية العرض .
- إلغاء محتوى الجهاز ومن ثم إنهاء المعالجة .

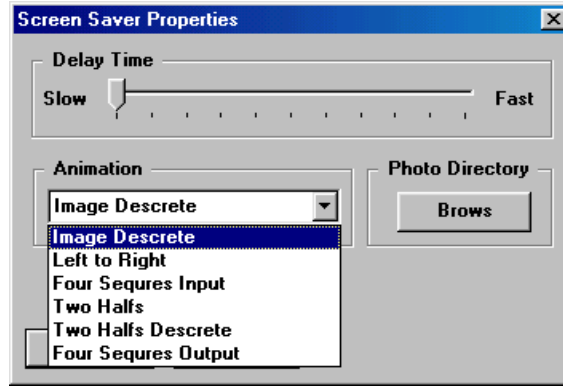
وتجدر الإشارة هنا إلى انه قبل خوارزمية العرض لابد من الحصول على الصورة النقطية ضمن المسار المحدد لعدد من الملفات الموجودة في دليل ما ومن ثم عرضها الواحدة تلو الأخرى. وعن طريق المتغير الموجود في صندوق الحوار Brows يتم تحديد المسار المطلوب ، لاحظ الشكل (8) .



الشكل (8) : تحديد دليل الملفات الصورية

بعد الانتهاء من هذه العمليات والتي عن طريقها تم التعامل مع واجهة الجهاز الرسومية، سيتم تحديد الخوارزمية المطلوبة حسب طلب المستخدم الذي سبق خزنه في المتغير chose ضمن التسجيل . الخوارزميات المتوفرة هي (الشكل 9):

الشكل (9) :
قائمة عرض
الخوارزميات



أولاً: خوارزمية (Image Discrete) ، عملية عرض للصورة تتم بأخذ بياناتها من محتوى الجهاز للذاكرة عن طريق تقسيم الصورة بأبعادها إلى عدد من المربعات الصغيرة إذ تعرض نقطة من كل مربع من المربعات ويشكل متزامن إلى أن تمتلئ المربعات في الوقت نفسه، وتتم هذه العملية :
لاحظ الملحق (1)

- تحديد المربعات المطلوبة لتقسيم الشاشة اعتماداً على حجم الصورة.
- اخذ البيانات من الصورة ووضعها في المواقع المحددة للمربعات .
- وضع البيانات على الشاشة نقطة نقطة.
- تكرار الخطوتين أعلاه الى ان تكتمل بيانات الصورة .

ثانياً: خوارزمية (Left to Right) ، يتم عرض الصورة بشكل تحديد الاتجاه من اليسار إلى اليمين أي العمود الأول ومن ثم الثاني إلى حين اكتمال الصورة على الشاشة . ويتم العرض عن طريق : الملحق (2).

- تثبيت المحور العمودي للصورة .
- التغيير يكون عن طريق المحور الأفقي .
- عرض كل نقطة للصورة.

ثالثاً: خوارزمية (Four Sequres Input) ، تقسيم الصورة إلى أربعة مربعات متساوية ويكون العرض متزامناً في المواقع الأربعة المحددة الى ان تكتمل الصورة على الشاشة وتتجز العملية عن طريق : الملحق (3).

- تثبيت النقطة الأولية لكل مربع والمأخوذ من بيانات الصورة .
- تكوين دائرة معينة وبعدها لا يتجاوز أبعاد الصورة ثم يتم العرض لكل نقطة في المربعات المحددة وفي الوقت نفسه.

رابعاً: خوارزمية (Two Halfs) ، تكون صيغة العرض بتحديد عمود من جهة اليسار و آخر من جهة اليمين ويتم رسم الصورة من الاتجاهين لحين الوصول إلى منتصف الشاشة عندها تكتمل الصورة، الملحق (4).

خامساً: خوارزمية (Two Halfs Discrete) ، يتم العرض فيها عن طريق تكوين خطوط أفقية تبدأ من اليسار والأخرى من جهة اليمين ولكن ليست على المستوى نفسه ويستمر العرض إلى حين اكتمال الصورة على الشاشة، لاحظ الملحق (5).

وهكذا فإن الصورة المستخدمة في الخوارزميات يتم الحصول عليها من خلال دالة البحث وذلك في المسار المأخوذ من قاعدة البيانات ، وعند عرضها يتم البحث عن أخرى وهكذا دواليك . إن العملية الأخيرة التي يجب القيام بها هي إلغاء الكائن الرسومي وإلغاء لمحتوى الأجهزة المعرفة وبعد ذلك يتم الانتهاء من الإجراء الرئيس للبرنامج .

ب) إجراء صندوق الحوار Screen Saver Configure Dialog

للقيام بعملية التغيير في إعدادات التطبيق يتم استخدام الدالة ScreenSaverConfigureDialog() والتغيير يتم عن طريق العرض لصندوق حوار التطبيق المكون في ملف المستلزمات والذي يشتمل على عدد من أشكال الأزرار والإدخالات (الشكل 3) بعد ان يتم اختيارها او تغييرها من قبل المستخدم مما يؤدي الى تغيير بيانات التسجيل في قاعدة بيانات التطبيق . ويشتمل هذا الإجراء على الرسائل الأساسية الآتية: لاحظ الملحق (6)

أولاً: رسالة ابتداء صندوق الحوار WM-INITDIALOG

تُرسل إلى صندوق الحوار مباشرة قبل ان يعرض او يظهر على الشاشة، اذ يستخدم صندوق الحوار هذه الرسالة لإعطاء قيم ابتدائية للمتغيرات الموجودة والتي سوف تظهر على الشاشة

ومن خلال هذه الرسالة تمت المعالجة الآتية على صندوق الحوار :

□ تكوين بعض الكائنات الرسومية التابعة لصندوق الحوار والتي لم نستطع تكوينها من خلال ملف المستلزمات كونها ليست رسومات قياسية ضمن ملف المستلزمات ، أي القيام بإضافة عدد من الإدخالات الرسومية على صندوق الحوار وباستخدام مكتبة خاصة وربطها بالتطبيق وهي COMCTL32.LIB فضلاً عن الملف التمهيدي .COMMCTRL.h.

□ عملية إظهار إعدادات التطبيق في صندوق الحوار عن طريق اخذ هذه الإعدادات من قاعدة البيانات الخاصة بالتطبيق ومن مفتاح التسجيل المتكون في الإجراء الأول للبرنامج، أي إظهارها في الكائنات الرسومية التابعة لكل إعداد أو متغير .

وصندوق الحوار هنا يمكن المستخدم الإبقاء على إعدادات التطبيق أو تغييرها حسب الرغبة .

ثانياً: رسالة الاوامر WM-COMMAND

تُرسل في حالة اختيار المستخدم لكائن معين من صندوق الحوار:

□ **Brows** : ستظهر نافذة صغيرة تحوي هيكلًا هرمياً لدلائل القرص الصلب (الشكل 8)، إذ يتم اختيار مسار لدليل معين من قبل المستخدم ثم غلق الشاشة والرجوع الى صندوق الحوار، وهذا الدليل سيثبت في قاعدة البيانات الخاصة بالتطبيق لتجري المعالجة عليه ضمن الإجراء الرئيس للبرنامج فضلاً عن عملية البحث عن ملفات الصور ذات الامتداد BMP و PCX. ليتم عرضها في التطبيق .

□ **Ok** : سيقوم التطبيق بحفظ جميع الإعدادات المدخلة من قبل المستخدم في صندوق الحوار ومن ضمنها الدليل المحدد ووضعها في قاعدة البيانات الخاصة بالتطبيق ، ثم غلق صندوق الحوار .

□ **Cancel** : يعمل على غلق صندوق الحوار من دون حفظ الإعدادات الجديدة للتطبيق. بمعنى آخر، ستبقى الإعدادات السابقة للتطبيق كما هي.

الاستنتاجات

التطبيق (ScreenSaver) هو قالب لحافظة شاشة يمكن أن تستخدم فيه أية صورة موضوعة ضمن أي دليل داخل الحاسبة ، كما يمكن استخدام إحدى خوارزميات العرض الخمسة للتقطيع والعرض لجعل الصور متوافقة مع الهدف من حافظة الشاشة ، أما الصور المستخدمة فيجب ان تحمل الامتداد BMP و PCX. وهي الأنواع المعتمدة في ويندوز . كما يمكن استخدام القالب أسلوباً لمنع غير المخولين من الدخول إلى الحاسبة دون إعطاء كلمة المرور المطلوبة . تم تنفيذ هذا الأسلوب الذي اثبت فاعليته على العكس من كلمات المرور الموجودة ضمن ويندوز التي غالباً ما يتم تجاوزها بأي شكل من الأشكال.

الملحق

(1) خوارزمية (Image Discrete)
 void **anim_1** (hdc hdc)
 {
 for (int f=0; f<4; f++)
 for (int c=0; c<4; c++)
 for (int y=f; y<bm.bmheight; y+=4)
 for (int x=c; x<bm.bmwidth; x+=4)
 setpixel (hdc, x, y, getpixel (hdcmemory, x , y));
 }

(2) خوارزمية (Left to Right)
 void **anim_2** (hdc hdc)
 { int tx = (bm.bmwidth - 1) / 2; int ls = tx + 1;
 for (int x=tx; x>0; x--, ls++)
 for (int y=0; y<bm.bmheight; y++) {
 setpixel (hdc, x, y, getpixel (hdcmemory, x , y));
 setpixel (hdc, ls, y, getpixel (hdcmemory, ls, y)); }
 }

(3) خوارزمية (Four Squares Input)
 void **anim_3** (hdc hdc)
 { int ls = bm.bmwidth-1; int tx = ls / 2;
 for (int x=0; ls>tx; x++, ls--)
 for (int y=0; y<bm.bmheight; y++) {
 setpixel (hdc, x, y, getpixel (hdcmemory, x , y));
 setpixel (hdc, ls, y, getpixel (hdcmemory, ls, y)); }
 }

(4) خوارزمية (Two Halfs)
 void **anim_4** (hdc hdc)
 { int i, j, k, l;
 setpixel (hdc, x1, y1, getpixel (hdcmemory, x1 , y1));
 setpixel (hdc, x2, y2, getpixel (hdcmemory, x2 , y2));
 setpixel (hdc, x3, y3, getpixel (hdcmemory, x3 , y3));
 setpixel (hdc, x4, y4, getpixel (hdcmemory, x4 , y4));
 for (i=1; i<y1+1; i++) { l=i;
 for (k=1; k<3; k++) {
 for (j=x1-i; j<x1+i+1; j++) {
 setpixel (hdc, j, y1-l, getpixel (hdcmemory, j, y1-l));
 }
 }
 }

```

        setpixel (hdc, j+x1*2, y1-1, getpixel (hdcmemory, j+x1*2, y1-1));
        setpixel (hdc, j, y1*3-1, getpixel (hdcmemory, j, y1*3-1));
        setpixel (hdc, j+x1*2, y3-1, getpixel (hdcmemory, j+x1*2, y3-1));
        } l=l*-1; }
    }
    for (i=1; i<x1+1; i++) { l=i;
        for (k=1; k<3; k++) {
            for (j=y1-i+1; j<y1+i; j++) {
                setpixel (hdc, x1+1, j, getpixel (hdcmemory, x1+1, j));
                setpixel (hdc, x1*3+1, j, getpixel (hdcmemory, x1*3+1, j));
                setpixel (hdc, x1+1, j+y1*2, getpixel (hdcmemory, x1+1,
                    j+y1*2));
                setpixel (hdc, x1*3+1, j+y2*2, getpixel (hdcmemory, x1*3+1,
                    j+y2*2)); }
            } l=l*-1; } }
    }

```

(5 خوارزمية (Two Halfs Discrete)

```

void anim_5 (hdc hdc)
{ int ls = bm.bmwidth-1; int tx = ls / 2;
  for (int x=0; ls>0; x++, ls--)
    for (int y=0; y<bm.bmheight; y+=2) {
        setpixel (hdc, x, y, getpixel (hdcmemory, x , y));
        setpixel (hdc, ls, y+1, getpixel (hdcmemory, ls, y)); }
    }

```

6 إجراء صندوق الحوار ScreenSaverConfigureDialog

```

bool WINAPI screensaverconfiguredialog
    (hwnd hdlg, uint message, WPARAM wParam, LPARAM lParam)
{
    switch (message) {
        case WM_INITDIALOG:
            initcommoncontrols (); isok = true;
            regcreatekeyex (hkey_local_machine, "software\\screen_saver",
                0, "screen saver", 0, key_all_access, null, &hregkey, &result);
            regqueryvalueex (hregkey, "delay", null, &datatype, null,
                &datasize);
            regqueryvalueex (hregkey, "delay", null, &datatype, lpbYTE)
    }
}

```

```

&delay, &datasize);
regqueryvalueex (hregkey, "chosse", null, &datatype, null,
&datasize);
regqueryvalueex (hregkey, "chosse", null, &datatype, (lbyte)
&chosse, &datasize);
regqueryvalueex (hregkey, "path", null, &datatype, null,
&datasize);
regqueryvalueex (hregkey, "path", null, &datatype, (lbyte)
&path, &datasize);
hwndslider = createwindow (trackbar_class, "",
ws_child|ws_visible|tbs_autoticks,50,30,240,25,hdlg,null,hd, null);
sendmessage (hwndslider, tbm_setrange, (wparam)1, (lparam)
makelong (0, 100));
sendmessage (hwndslider, tbm_setpos, (wparam>true, (lparam)
(delay/10));
sendlgitemmessage (hdlg, idc_combo1, cb_addstring, 0,
(lparam) "image discrete");
sendlgitemmessage (hdlg, idc_combo1, cb_addstring, 0,
(lparam) "left to right");
sendlgitemmessage (hdlg, idc_combo1, cb_addstring, 0,
(lparam) "four seques");
sendlgitemmessage (hdlg, idc_combo1, cb_addstring, 0,
(lparam) "two halves");
sendlgitemmessage (hdlg, idc_combo1, cb_addstring, 0,
(lparam) "two halvesdiscrete");
sendlgitemmessage (hdlg, idc_combo1, cb_addstring, 0,
(lparam) "zzzzzzzzzzzz");
sendlgitemmessage (hdlg, idc_combo1, cb_setcursel, (((chosse
>= 1) && (chosse <= 6)) ? chosse - 1: 0), 0);
return true;
case wm_command:
switch (loword (wparam)) {
case idset: showbrowse (hdlg);
isok = showfile ("\\*.bmp"); break;
case idok:
if (!isok) { killtimer (htimer, 1);
messagebox (null, msg, "warring", 0); }
else {

```

```
sp = sendmessage (hwndslider, tbn_getpos, (wparam)0,
                  (lparam)0);
killtimer (htimer, 1);      delay = sp * 10;
timer = settimer (htimer, 1, delay, null);
chosse = senddlgitemmessage (hdlg, idc_combo1, b_getcursel,
                             0, 0) + 1;

if (chosse == 0)    chosse=1;
regsetvalueex (hregkey, "delay", 0, reg_dword, (labyte)
               &delay, sizeof (dword));
regsetvalueex (hregkey, "chosse", 0, reg_dword, (labyte)
               &chosse, sizeof (dword));
regsetvalueex (hregkey, "path", 0, reg_sz, (labyte)path, strlen
               (path)+1);
}
regclosekey (hregkey);    enddialog (hdlg, 0);    break;
case idcancel:    enddialog (hdlg, 0);    return true;
}    return true;
}    return false;
}
```

المصادر

- [1] Andrews, M., 1996, “**Learn Visual C ++ Now**”, *Ms Press*.
- [2] Myers, B. & Doner, C., 1988, “ **Graphics Programming under Windows**”, *SYBEX Inc*.
- [3] Pappas, C., H. & Murray, W., H., 1997, **The Complete Reference Visual C++ 5**, *McGraw-Hill Companies*.
- [4] Petzold, C., 1992, **Programming Windows 3.1**, *3rd Edition*, *Microsoft Press*.
- [5] Randall, N., 2000, “Give Your Desktop a Makeover: Computer Programs”, *PC Magazine*, *Vol.19, Issue6, P.104*.
- [6] Rubenking, N.J, 2000, “Mutate your Screen: Screen Savers”, *PC Magazine*, *Vol. 19, Issue 16, , P. 117*.
- [7] Schildt, H.,1998, **Windows 98 Programming from the GROUND UP**, *McGraw-Hill Companies*.
- [8] Steers, K., 1999, “Troubleshooting: Screen Saver”, *PC World*, *Vol. 17, No. 8, P.136*.
- [9] Steers, K., 2000, “Windows System Monitor Peeks Under a PC Hood : Screen Savers”, *PC World*, *Vol. 18, Issue 11*.
- [10] Yao, P. & Leinecher, R., C., 1997, “**Visual C++ 5 Bible**”, *IDG Books World Wide Inc*.