Projet : Générateur de Codes QR

Cahier des charges



# Notes d’édition

Première édition @2024

Création de l’œuvre : Raphaël Tremblay-Lessard

Collaboration(s) : Patrick Simard (document modèle)

Actualisation(s) : Aucun

## Conditions d’utilisation

L’auteur autorise les enseignants du département d’informatique du Cégep de Chicoutimi à utiliser cette œuvre exclusivement dans le cadre de leur cours de la formation régulière en présentielle au même collège.

En cas d’appropriation d’une portion mineure afin de contribuer à la création d’une nouvelle œuvre ou en ajout à une œuvre existante, l’auteur demande que sa participation soit reconnue et qu’il soit mentionné dans la nouvelle œuvre à titre de collaborateur. La consultation de l’auteur pour cet acte est fortement suggérée.

L’appropriation d’une portion majeure afin de contribuer à la création d’une nouvelle œuvre ou en ajout à une œuvre existante n’est pas autorisée par l’auteur. L’auteur demande à être consulté au préalable afin d’obtenir son autorisation.

En cas de modification mineure (ajout mineur, retrait ou modification d’élément), l’auteur invite l’enseignant concerné à le mentionner à la section *Actualisation(s)* en ajoutant son nom et la ou les année(s) où cette action a été posée.

## Reconnaissance et protection du droit d’auteur

Selon l’article 8-1.03 de la convention collective des enseignants, les cahiers de notes ou notes de cours ne peuvent être utilisés sans le consentement explicite de l’auteur. De plus, selon l’annexe V-4 la reproduction sous une forme quelconque, la présentation en public et la publication d’un œuvre est un droit exclusif appartenant à l’auteur.

[Source de l'image de la page de garde : www.louisianafcu.org](https://www.louisianafcu.org/articles/the-polite-persons-guide-to-splitting-the-bill)

Table des matières

Présentation du projet 1

Contexte 1

Besoin 1

Exemple d’utilisation simple 1

Description de la demande 2

Objectifs 2

Fonctionnalités du système 2

Critères d’acceptabilité 3

Contraintes du projet 4

Contraintes des coûts 4

Contraintes de délais 4

Solutions proposées 5

Solution A – Application web 5

Solution B – Application native avec base de donneés 5

Solution C – Application native sans base de donneés 5

Solution recommandée 6

Solution recommandée 6

Argumentation 6

Déroulement du projet 7

Planification 7

Spécifications techniques 7

Langage de développement 7

Patron de conception 7

Sauvegarde des données 7

Devises 7

# Présentation du projet

## Contexte

Ce projet porte sur le besoin exprimé par une entreprise de marketing qui veut générer des codes QR personnalisés pour ses clients.

## Besoin

À cette fin, cette entreprise a besoin d’une application qui permet de créer différents codes QR. Cette application devra être séparée en deux parties principales : le générateur de code QR, et l’interface de préparation et de visualisation.

Entre autres, l’entreprise souhaite être capable de générer des codes QR **personnalisés pour la clientèle** à partir de différents types des données (texte, adresses web, images, etc.) et exporter le résultat dans des fichiers d’image .png.

## Exemple d’utilisation simple

Une pizzéria locale veut inclure un code QR pointant vers son site web dans sa campagne de marketing. Après que le projet est passé par un graphiste et un publiciste, un employé utilise le logiciel de création de codes QR pour transformer l’adresse <http://pizzasaglac.com> en incluant une petite image de pointe de pizza dans le coin de l’image générée.

# Description de la demande

## Objectifs

L’objectif principal du projet consiste à développer une application de génération de codes QR qui devra permettre de réaliser les tâches suivantes :

* Générer des codes QR respectant le standard
* Incruster une image dans le code QR généré
* Utiliser une autre image plutôt que les « pixels » d’un code QR typique.
* Sauvegarder et récupérer les données

## Fonctionnalités du système

Afin d’effectuer adéquatement les tâches précédentes, voici les différentes fonctionnalités qui devront être offertes par le système :

* Accepter des messages de longueur variable
* Accepter différents types des données (texte, adresses web, etc.)
* Encoder les données à l’aide de l’algorithme de correction d‘erreurs selon le niveau de tolérance choisi.
* Disposer l’information dans une matrice de pixels respectant le standard, incluant les méta-informations nécessaires.
* Appliquer un masque sur le code généré pour améliorer la lisibilité.
* Exporter le résultat dans des fichiers d’image .png
* Identifier les sections « sacrifiables » du code QR pour placer l’image personnalisée

## Critères d’acceptabilité

Il est entendu avec les clients que le livrable final de l’application respectera les critères d’acceptabilité suivants :

* L’application permet de réaliser adéquatement les objectifs nommés ci-haut.
* L’application sera séparée en deux programmes principaux : le générateur de code QR, et l’interface de préparation et de visualisation.
* L’application console générant les codes QR n’aura aucune interface visuelle, prendra son information à la ligne de commande, et produira des fichiers .png en retour.
* L’interface de préparation et de visualisation permettra à l’utilisateur d’entrer dans une interface pratique l’information destinée à l’application console. De plus, elle permettra de visualiser le résultat du travail de cette invocation.
* La conception de l’interface est ergonomique, professionnelle, simple à utiliser et s’accorde aux standards des applications que l’on rencontre habituellement dans l’environnement Windows.
* L’interface devra permettre un ajustement de la taille de la fenêtre principale selon le souhait de l’utilisateur tout en conservant une disposition fonctionnelle des composants.
* **Un guide d’installation et un guide d’utilisation accompagneront le livrable final.**

# Contraintes du projet

## Contraintes des coûts

Étant donné qu’il s’agit d’un travail scolaire, aucun coût n’est réellement chiffrable.

Toutefois, le seul coût que l’on pourrait envisager pour ce projet consisterait au salaire des employés, soit les heures de travail \* taux horaire des employés. Par conséquent, on peut déterminer la contrainte des coûts à l’aide du temps à investir par les membres de l’équipe. Ainsi, il en coûtera 35$ pour 15 heures de travail par semaine durant 9 semaines pour chaque membre de l’équipe.

Soit environ 675 heures par équipier.

## Contraintes de délais

Le projet devra être terminé pour le 29 mars 2024

# Solutions proposées

## Solution A – Application web

### Avantages

* Accessible de partout dans le monde
* Accessible par tous les participants
* Accessible à l’aide de n’importe quel dispositif électronique ayant accès à internet et d’un navigateur web

### Inconvénients

* Requiert d’être hébergé avec des frais associés
* Peut ne pas être bien supporté par certains navigateurs
* Requiert une gestion de droits ou d’authentification

## Solution B – Application native avec base de donneés

### Avantages

* Ne requiert pas d’accès à internet
* Dédié à une utilisation sur le système d’exploitation ciblé

### Inconvénients

* Requiert une gestion de droits ou d’authentification
* Selon la configuration locale ou en réseau du système de gestion de bases de données (SGBD), l’application sera accessible par une instance locale ou plusieurs instances sur le réseau

## Solution C – Application native sans base de donneés

### Avantages

* Ne nécessite pas une gestion de droits ou d’authentification
* Dédié à une utilisation sur le système d’exploitation ciblé
* Portabilité et distribution facile de l’application
* Utilisation indépendante facilement par différents tiers

### Inconvénients

* Accessibilité limitée aux données à un seul poste dédié

# Solution recommandée

## Solution recommandée

La **solution C**, ***application native sans base de données***, est recommandée et fait l’objet de la suite de ce document.

## Argumentation

Cette solution est particulièrement intéressante pour le client puisqu’il s’agit d’une utilisation interne pour une petite entreprise.

Comme il ne leur est pas essentiel de disposer de plus d’accessibilité que ce qu’offre cette solution, il est donc aussi approprié de ne pas complexifier l’accès à l’application et aux donnés. Ceci tant pour le développement que pour le déploiement de la solution et que pour sa configuration.

Le développement de la solution n’aura pas à intégrer une technologie tierce pour l’enregistrement des données ce qui va ainsi limiter les soucis d’intégration et de stabilité lors de l’utilisation.

Enfin, la grande simplicité d’installation et de portabilité de la solution permettra une prise en main facile et rapide par quiconque est suffisamment habile à manipuler des fichiers sur un ordinateur.

# Déroulement du projet

## Planification

Afin de respecter la contrainte de délais, voici de manière détaillées les différents jalons de ce projet.

|  |  |
| --- | --- |
| **Livrables** | **Échéance** |
| Cahier de charges fonctionnel | 26 janvier 2024 |
| Conception détaillée de la preuve de concept | 2 février 2024 |
| Conception détaillée du projet final | 23 février 2024 |
| Application :  -preuve de concept  -produit final | 23 février 2024  22 mars 2024 |
| Guide d’installation et guide d’utilisateur | 29 mars 2024 |
| Bilan de projet(post-mortem) | 29 mars 2024 |
|  |  |

## Spécifications techniques

### Langage de développement

### Nous avons choisi C#.

### Patron de conception

L’application sera séparée en deux programmes principaux : le générateur de code QR fait en console, et l’interface de préparation et de visualisation fait avec MVVM.

### Sauvegarde des données

### Selon la solution recommandée il n’y aura aucune sauvegarde de données nécessaire.

### Devises

« La clé est dans le détail. » - Jacques-Antoine Granjon (Vente-Privée.com)

Nous nous efforçons toujours d’avoir un plan minutieux qui est réfléchi et cela se reflète sur la satisfaction de notre clientèle grandissante.