**Guia do Apostador**

Versão <1.0>**Controle de Versão**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versão** | **Data** | **Razões para alteração** | **Responsável** |
| 1.0 | 24/03/2015 | Documento Inicial | Claudio Sá Junior |
| 1.1 | 24/03/2015 | Complemento do Documento | Sérgio Prates / Victor Alencar |

**Envolvidos na elaboração do Documento de Arquitetura**

|  |  |
| --- | --- |
| **Nome** | **Área** |
| Claudio Sá Junior |  |
| Gabrielle Macedo |  |
| Sergio Prates |  |
| Victor Alencar |  |

**Índice**

# Introdução

## Objetivo do Documento

Este documento apresenta uma visão geral abrangente da arquitetura do projeto Guia do Apostador gerado pela área de Projetos, utilizando uma série de visões arquiteturais diferentes para ilustrar seus diversos aspectos.

Sua intenção é capturar e transmitir as decisões significativas do ponto de vista da arquitetura que foram tomadas em relação ao projeto.

## Escopo

O Projeto Guia do Apostador tem como finalidade auxiliar os apostadores de loterias à consultarem os resultados dos concursos de maneira mais rápida e confortável, ao invés do atual modelo no qual os apostadores têm que ir na lotérica e consultar os números que foram sorteados.

## Siglas, Abreviações e Acrônimos

Framework – Artefato que auxilia no desenvolvimento de algo específico, pode auxiliar tanto no desenvolvimento front-end quanto back-end.

Front-end – Camada de visualização, pode ser um website, um aplicativo ou qualquer aplicação que seja visualizável.

Back-end – Camada de lógica, esta camada não fica acessível para o usuário, nela contém por exemplo regras de acesso ao banco de dados, cálculos e etc.

Multi-plataforma – Algo que pode ser executado em ambientes distintos, por exemplo, uma aplicação multi-plataforma pode ser executada tanto em ambientes Windows, Mac ou Linux.

## Referências

1. <http://www.highcharts.com/docs> - Componentes gráficos
2. <http://docs.phonegap.com/en/4.0.0/index.html> - Framework javascript que permite o acesso a recursos nativos de aparelhos móveis em sistemas operacionais distintos.
3. <https://msdn.microsoft.com/en-us/library/aa287558(v=vs.71).aspx> – Linguagem de desenvolvimento back-end.
4. <http://www.agileandart.com/2010/07/16/ddd-introducao-a-domain-driven-design/> -Pattern de Desenvolvimento
5. <https://angularjs.org/> - Framework para desenvolvimento Front-end baseado em MV\*.
6. <http://getbootstrap.com/> - Framework para desenvolvimento Front-end
7. DocumentoDeRequisitos.pdf, versão 1.0 de 11/03/2015 em https://github.com/sergioprates/guiadoapostador/tree/master/Documentacao/Analise
8. VisaoDasNecessidades.docx, versão 1.0 de 04/03/2015 em https://github.com/sergioprates/guiadoapostador/tree/master/Documentacao/Analise
9. http://jqueryvalidation.org/ - Framework para validação de formulários.
10. https://jquery.com/ - Framework para desenvolvimento front-end.

# Metas e Restrições da Arquitetura

Para a proposta da arquitetura, foram considerados alguns fatores como a finalidade da aplicação, o tipo de usuário em questão e os ambientes em que a plataforma irá operar como um todo. A arquitetura deverá então atender aos seguintes requisitos:

* Modularidade: O sistema deverá possuir suas partes desacopladas, o que facilitará a reutilização de regras de negócio já construídas em outras aplicações caso seja necessária.
* Versão do .NET Framework: 4.0
* Versão do phonegap: 3.7.0
* Visual Studio 2010: IDE de desenvolvimento .NET
* Intel XDK: IDE que auxilia no desenvolvimento de apps híbridos.
* Utilização de componentes e frameworks de terceiros

1. Log4net: Lib criada para gerar logs do sistema, será utilizada na maioria das vezes para tratar exceções.

* SqlHelper: Lib para facilitar acesso a dados.
* SQL Server 2012: Banco de dados que será utilizado.
* HTML5, js e css3: Interface para o usuário.
* Processos Batch

1. Será implementado um processo para fazer uma carga inicial dos dados. Os dados serão consultados via serviço e inseridos na base de dados da aplicação.
2. Terá outro processo que irá rodar diariamente para atualizar os dados de sorteios novos e trazer para a base de dados da aplicação.

* Envio de email: Quando o apostador acertar algum número do sorteio, o sistema deverá enviar um e-mail informando e incentivando-o a apostar novamente.

# Visão de Casos de Uso

[Esta seção lista os casos de uso ou cenários do modelo de casos de uso se eles representam uma funcionalidade central e significativa do sistema final ou se têm uma ampla cobertura de arquitetura, ou seja, se experimentam muitos elementos arquiteturais ou se enfatizam ou ilustram um determinado ponto frágil da arquitetura. Exemplos: Um caso de uso que requer a transmissão de imagens, um caso de uso que requer a integração com outra aplicação.]



**Gerar Sorteio**

**<<include>>**

## Ações do Apostador

1. Consultar Sorteio: O apostador pode consultar sorteios já realizados.
2. Manter Aposta: O apostador poderá inserir suas gerenciar suas futuras apostas.
3. Gerar Sorteio: Terá uma funcionalidade que o programa gerará um sorteio para o usuário, que poderá ou não definir como seu jogo.
4. Consultar Estatísticas: Terá diversas estatísticas dos jogos realizados da própria CAIXA e dos jogos salvos do usuário.

# Visão Lógica

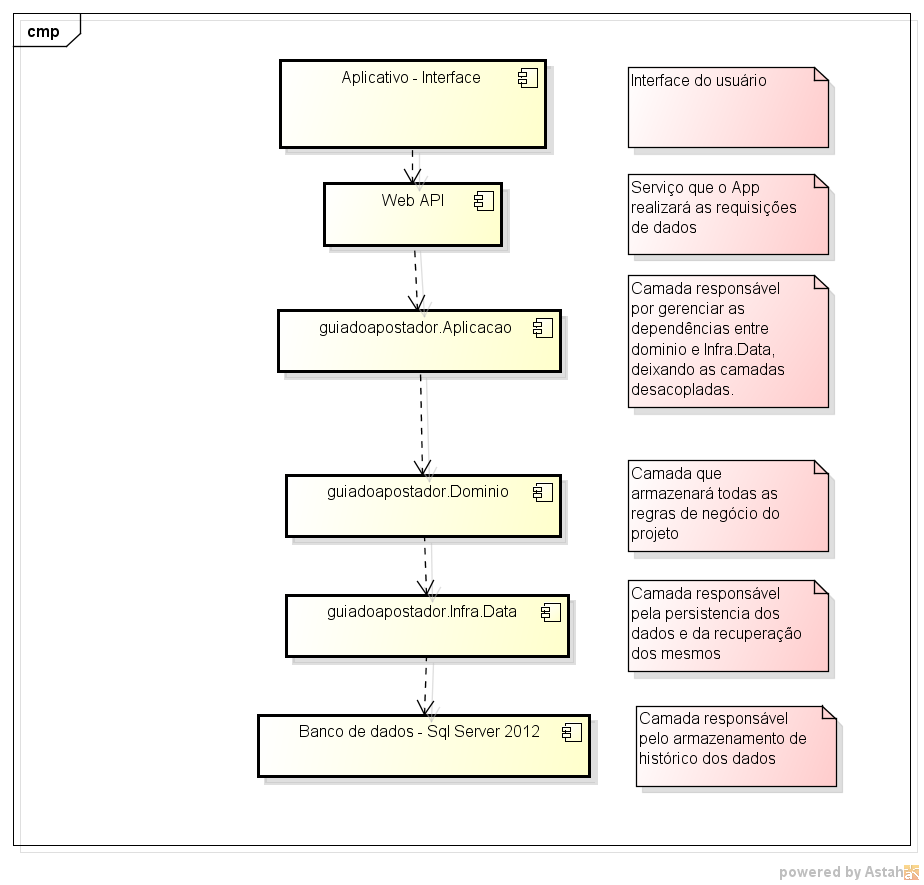
## Visão Geral

< ILUSTRAÇÃO DE ALTO NÍVEL COM PRINCIPAIS ELEMENTOS DA SOLUÇÃO >

[Descrição dos elementos da ilustração e citação dos casos de uso que eles representam.]

## Camadas da Aplicação

Para este projeto será implementado o design pattern DDD que visa o máximo de desacoplamento de código possível, utilização de Injeção de Depências, não acoplando as classes diretamente umas as outras.



## Nomenclatura da Aplicação

A aplicação a ser desenvolvida terá as seguintes identificações:

|  |  |
| --- | --- |
| Macrosistema | Guia do Apostador |
| Nome da aplicação | guiadoapostador |
| Pacotes | guiadoapostador.API  guiadoapostador.Aplicacao  guiadoapostador.Dominio  guiadoapostador.Infra.Data |

## Componentes e frameworks a serem construídos

| Identificação | Responsável | Descrição |
| --- | --- | --- |
| phonegap | Desenvolvimento | Criar arquitetura de desenvolvimento de acordo com os outros frameworks. |
| Angularjs | Desenvolvimento | Dividir páginas em escopos distintos e realizar requisições para a API na aplicação phonegap. |
| Log4Net | Desenvolvimento | Implementar logs nas partes de alto risco do sistema. |
| jQuery | Desenvolvimento | Utilizar para validações |
|  |  |  |

## Componentes e frameworks a serem reutilizados

| Identificação | Responsável | Descrição |
| --- | --- | --- |
| Guiadoapostador.Dominio | Desenvolvimento | Biblioteca que será reutilizada ao ser implementado o processo batch na aplicação. Caso a aplicação venha a evoluir futuramente tornando-se, por exemplo, uma aplicação web, este componente será reutilizado para as ações do apostador. |

## Tratamento de Erros e Exceções

Todos os erros tratados serão logados em arquivos .log no servidor da aplicação. Serão pré-definidos alguns erros como “Sem conexão com a internet”, que serão apresentados de forma “amigável” ao usuário.

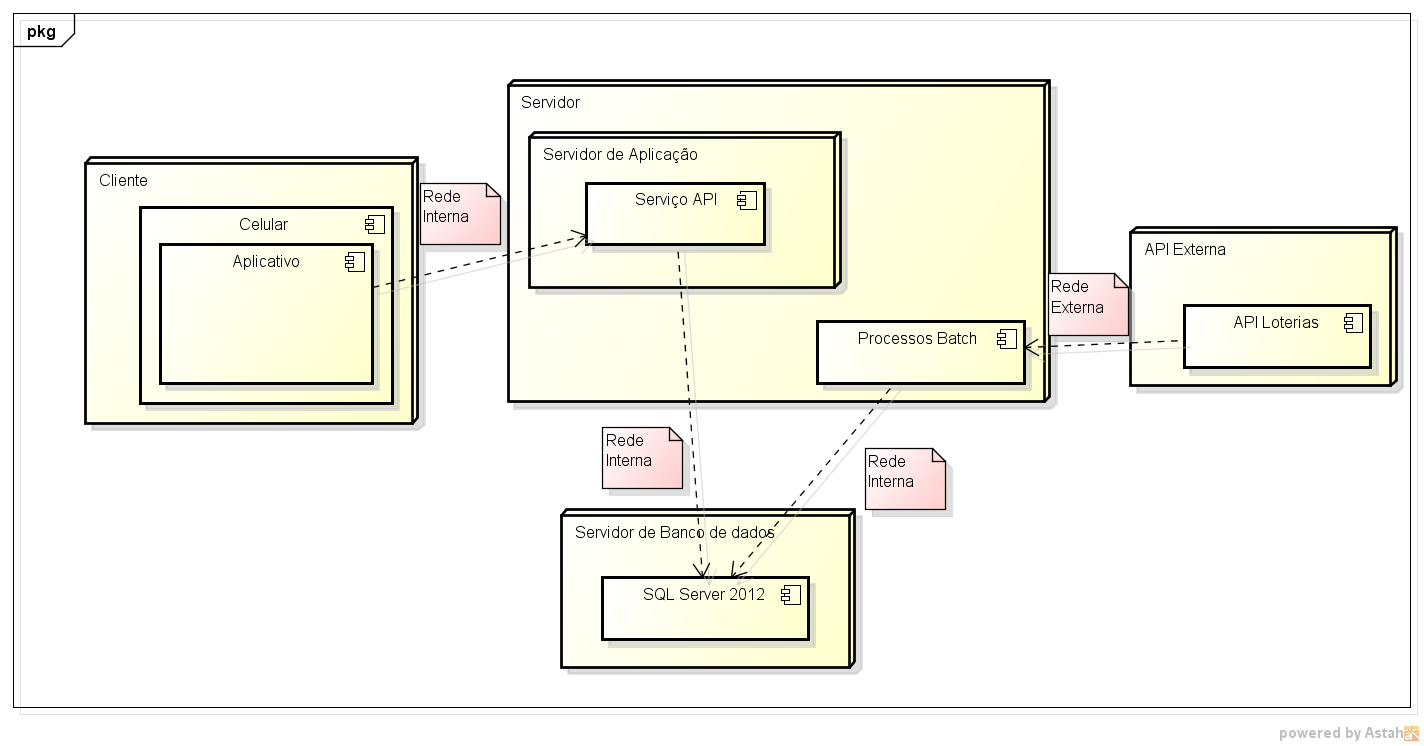
# Visão de Segurança

A aplicação terá as seguintes formas de acesso:

| Usuários | Tipo | Portal | Acesso |
| --- | --- | --- | --- |
| Apostador | Prestadores | Portal de Negócios | Externo |

A identificação dos apostadores será feita através do ID do dispositivo móvel.

# Visão de Implantação



## Servidor de Aplicações

| Datasource | Provider | XA | Versão | Servidor | Banco | Login |
| --- | --- | --- | --- | --- | --- | --- |
| Servidor\SQLSERVER2012 | SqlServer | NÃO | 11.0 | Zapdos | guiadoapostador | ASPNET |

## Servidor de Banco de Dados

Banco de dados: guiadoapostador

Endereço do Servidor: À definir

Usuário: ASPNET

Senha: gu1@d0@p0st@d0r

Collation: Default do SQL Server

# Tamanho e Desempenho

[Uma descrição das principais características de dimensionamento do software que têm um impacto na arquitetura, bem como as restrições do desempenho desejado. Corresponde a requisitos não funcionais descritos no Documento de Visão.]

* tempo de resposta de uma realização de caso de uso
* quantidade de usuários simultâneos da aplicação
* disponibilidade da aplicação
* frequência de mensagens trafegadas em uma integração (qtde / tempo)
* tamanho das mensagens trafegadas em uma integração (parâmetros ou bytes)
* frequência de imagens enviadas para o sistema de imagens (qtde / tempo)
* tamanho das imagens enviadas para o sistema de imagens (bytes)