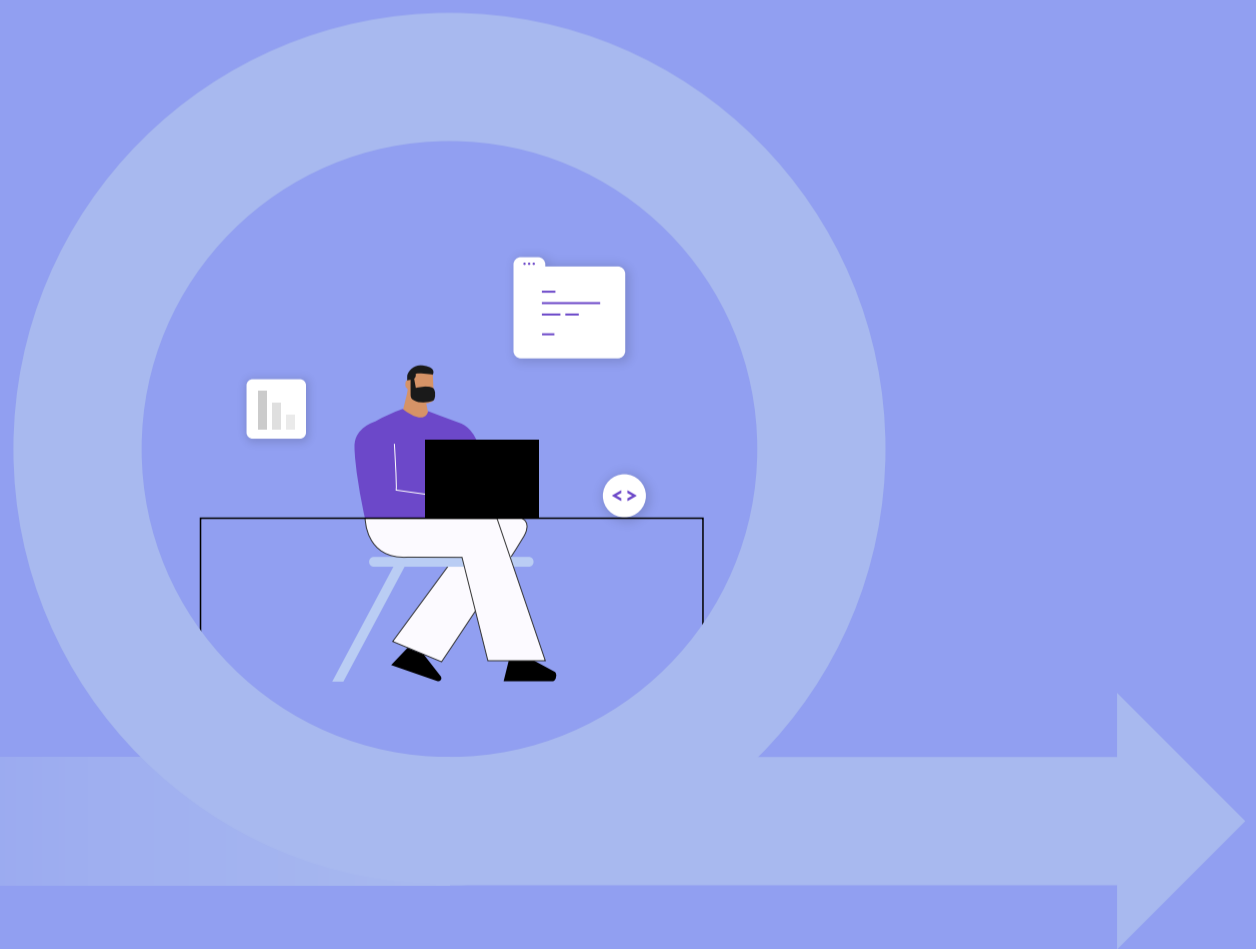


Japão

Mitsubishi Electric Corporation, Kamakura Works

Whitepaper



GeneXus™



Por que o Departamento de Sistemas da Informação, que costumava desenvolver sistemas totalmente internos, focou sua atenção no desenvolvimento low-code e decidiu adotá-lo?

A Mitsubishi Electric Corporation, Kamakura Works (doravante “Kamakura Works”) tem desenvolvido internamente todos os sistemas da informação utilizados no escritório, aproveitando ao máximo da mais recente tecnologia disponível. Consultamos seus membros sobre como introduziram GeneXus, um método de desenvolvimento low-code, e G.RAD.E para aumentar ainda mais a produtividade em 2018.

Todos os sistemas foram desenvolvidos internamente, exceto o sistema comum da empresa:

A Seção de Sistemas da Informação do Departamento de Gestão da Manufatura da Kamakura Works, cujos membros falamos nesta ocasião, é responsável pela manutenção e gestão de todos os computadores, servidores e redes utilizados na fábrica, bem como pelo desenvolvimento e manutenção da maioria dos sistemas utilizados nele. A Mitsubishi Electric Information Network Co., Ltd. é reconhecida não apenas como integradora de sistemas da Mitsubishi Electric, mas também como **integradora de sistemas da informação para uma ampla gama de indústrias.**

A Kamakura Works não é uma divisão que fabrica apenas produtos Mitsubishi Electric. Possui grande independência do escritório central e a estrutura é muito semelhante à de qualquer outra empresa, incluindo a gestão de pedidos do departamento de vendas e a gestão de produção e tarefas de pessoal, por isso esta seção se desenvolve e faz a manutenção de todos os sistemas da informação utilizados na Kamakura Works, além dos comuns a toda a empresa.

Naturalmente, estamos constantemente melhorando e refinando o sistema com base em propostas em cooperação com cada departamento para melhorar as operações comerciais.



Seção de Sistemas da Informação, Departamento de Gerenciamento de Manufatura, Kamakura Works, Mitsubishi Electric Corporation (da esquerda) Hiromi Higashi, Líder de Equipe, Toshiaki Nakamura e Shimpei Katsumoto, Chefe de Seção

Problemas surgidos no departamento que escolheu a base de desenvolvimento mais moderna:

Já que pensamos e promovemos soluções de negócios por conta própria, ao invés de deixá-las nas mãos dos fornecedores, sempre buscamos e selecionamos a melhor solução de infraestrutura e linguagem de desenvolvimento para se adequar ao sistema que vai ser desenvolvido.

Conseqüentemente, alguns desses sistemas, que eram originalmente de última geração, tornaram-se sistemas legados com o avanço da tecnologia. Recentemente, a falta de suporte para o Oracle Developer 2000, que usamos desde 2000, tem sido um problema.

Também temos desenvolvido com Java, mas para melhorar ainda mais a produtividade, investigamos o desenvolvimento low-code, no qual nos concentramos há muito tempo.



Mitsubishi Electric Information Network Corporation (da esquerda) Mr. Toyotaka Nakamura, Mr. Tomoaki Takano, Mr. Gen Ohno, Mr. Hideki Enomoto.

Escolha de GeneXus entre uma variedade de plataformas de desenvolvimento low-code:

“A partir da informação que encontramos na Internet, escolhemos uma plataforma de desenvolvimento que cumprisse as limitações de segurança inerentes a Kamakura Works, como a impossibilidade de utilizar a nuvem, e no final GeneXus e outra ferramenta de modelagem visual foram ambos finalistas. O fator decisivo foi o sistema de licenciamento.

GeneXus cobra por cada licença de desenvolvimento, enquanto as ferramentas de modelagem visual cobram por cada programa desenvolvido”, disse. Para nós, que desenvolvemos muitos sistemas, GeneXus, que cobra pelo número de desenvolvedores, é mais econômico.

A capacidade de reutilizar SQL, os procedimentos armazenados e outros ativos do sistema atualmente em uso também foi um ponto chave.

Além disso, a liberdade de escolha do sistema operacional e da linguagem foram fatores importantes. Também foi bom ter suporte multiplataforma e a possibilidade de escolher a linguagem a gerar. A sustentabilidade de poder migrar para um novo ambiente com o mesmo código-fonte, sem ser sobrecarregado por futuras alterações no sistema operacional ou no banco de dados, é muito atraente.

Com a mistura de várias versões de bancos de dados e sistemas operacionais, não é prático modificar cada programa individualmente, pois daria muito trabalho.

«GeneXus cobra por cada licença de desenvolvimento, enquanto as ferramentas de modelagem visual cobram por cada programa desenvolvido. Para nós, que desenvolvemos muitos sistemas, GeneXus, que cobra pelo número de desenvolvedores, é mais econômico.»

Estudo de templates para aumentar ainda mais a produtividade:

A Kamakura Works adotou GeneXus em 2018. No ano seguinte, iniciamos formalmente o desenvolvimento do sistema e, a partir de julho de 2020, lançamos sete sistemas.

Na ocasião, pudemos trocar ideias com outra empresa usuária em um evento de um fornecedor de ferramentas denominado GeneXus Day.

Esta empresa nos aconselhou que poderíamos aumentar nossa produtividade utilizando um produto de templates como WorkWithPlus ao invés de somente GeneXus. Pedimos então à empresa que explicasse a WeING Co., Ltd., que havia

solicitado apoio para o desenvolvimento com GeneXus.

Naquela época, WeING Co., Ltd. acabava de lançar um produto de infraestrutura de template chamado **G.RAD.E** (GeneXus Rapid Application Development Environment) usando WorkWithPlus, e recebeu esta introdução.

Estávamos interessados no ambiente de desenvolvimento G.RAD.E, sua aplicabilidade prática e sua melhoria de produtividade, e decidimos avaliá-lo em um negócio que estava programado para ser desenvolvido em Java na época.

Aumento real da produtividade em G.RAD.E e desafios para o futuro:

A tabela 1 mostra os resultados da avaliação do sistema que foi planejado para ser lançado em JAVA e desenvolvido em G.RAD.E. **A tabela 1** mostra os resultados da avaliação de cada tópico.

Quando se trata da folha de avaliação, algumas das telas são mais produtivas e alguns dos recursos são menos produtivos.

Como o desenvolvemos sem conhecimento prévio, demoramos um pouco para aprender e nos acostumar, mas isso está incluso no tempo de desenvolvimento na tabela.

Acreditamos que alguém mais familiarizado com o produto poderia ter feito isso de maneira mais eficaz.

Embora a migração do sistema atual para GeneXus exatamente da mesma maneira demore muito, funcionará melhor se for desenvolvido utilizando os templates fornecidos.

Se você não estiver familiarizado com a escolha de um modelo, haverá retrabalho e tentativa e erro. **Percebo que, assim que me acostumar, minha produtividade vai melhorar.**

Pessoalmente, temos a impressão de que será pelo menos 50% menor que em Java. É conveniente ser capaz de criar automaticamente partes do código que teriam que ser implementadas em Java, especialmente a parte de saída padrão.

Perguntamos a eles sobre os desafios de usar G.RAD.E. Foi difícil criar um ambiente com as limitações de segurança próprias da Kamakura Seisakusho. É necessário

padronizar os procedimentos de desenvolvimento e liberação, mas devemos nos preparar para o ambiente de desenvolvimento G.RAD.E. Neste sentido, estamos buscando uma forma de mudar a abordagem e a metodologia tradicional de desenvolvimento em cascata para uma mais adequada para GeneXus + G.RAD.E. Além disso, estabelecer uma metodologia para o desenvolvimento da equipe é um desafio para o futuro.

Table 1

| No. | Tela de destino | Tipo de tela | Horas de trabalho estimado por Java (pessoa - dias) | Número de horas de trabalho (pessoa - dias) |
|-----|--|--|---|---|
| 1 | Pesquisa de histórico de pagamentos | Lista de Pesquisa Tela principal-colegial | 5.5 | 1 |
| 2 | Manutenção do mestre do número do local | Magistral Lista de Pesquisa | 4.5 | 0.3 |
| 3 | Menu do sistema de solicitação de layout PWB | cardápio | 1 | 0 |
| 4 | Lista de dependências de layout PWB | Lista de Pesquisa | 5 | 4 |
| 5 | Dependências de layout PWB | Detalhes Lista de Pesquisa | 15 | 13 |
| 6 | Tela de anexo (Dependências de layout PWB) | Subtela | 5.5 | 1 |
| 7 | Tela de seleção do nome do departamento (Dependências de layout PWB) | Subtela | 5 | 1 |
| 8 | Mostrar comentários anteriores (Dependências de layout PWB) | Subtela | 3 | 1 |
| | | | 44.5 | 21.3 |

Descubra como GeneXus pode fazer o mesmo pela sua empresa.

info@genexus.com

