

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Danilo Silva Vargas
Lucas Carvalho Batista Canhadas Genvigir
Matheus Faustino Meneguim

PROJETO 2: INTEGRAÇÃO KODULAR E XANO

CORNÉLIO PROCÓPIO
2023

INTRODUÇÃO

Este documento explica as principais funcionalidades de um aplicativo (Aplicativo de Lista de Compras), desenvolvido com a plataforma Kodular, para a realização da lógica e interface gráfica em conjunto com o banco de dados do XANO(2023). O desenvolvimento deste aplicativo utilizou o conteúdo da disciplina Desenvolvimento sem Código, ministrada pelo Professor Matheus Mollon.

Todo o material de aula, disponível no Moodle da disciplina, foi utilizado como base de consulta. Parte de alguns blocos foram adaptados para o aplicativo.

1 DESENVOLVIMENTO DO APP DA LISTA DE COMPRAS

1.1 – Acesso ao Aplicativo

e-mail: matheus@email.com

senha: 12345

1.2 - Kodular

Após a inicialização da tela do aplicativo (quando o bloco *Card_entrar*¹ for inicializado), o bloco “*Web1*”² é chamado na sequência.

O bloco “*Card_entrar*” é referente ao “*botão entrar*” e foi utilizado para inicializar o banco de dados dentro do código, inicializando as variáveis de e-mail e senha no mesmo. A partir do if dentro deste bloco, dá início nas verificações de e-mail e senha, onde há a primeira verificação para que todos os campos não podem estar vazios para que assim ocorra o armazenamento das variáveis e mostra para o usuário circular progress, logo após vem as notificações que podem aparecer que são, se o usuário deixar todos os campos em branco, apenas a senha ou apenas o e-mail, essas verificações acabam no bloco “*Web1*”. O bloco “*Web1*” como mostrado na Figura 1, começa mudando o “*Circular_Progress1*” para não visível ao usuário, logo após verifica se os dados digitados pelo usuário (e-mail e senha) são verdadeiros retornando o código 200

¹ O bloco *Card_entrar* não está neste texto, deve ser visualizado no Kodular.

² O bloco *Web1* não está neste texto, deve ser visualizado no Kodular.

(código 200 é a resposta de sucesso, que a requisição foi bem sucedida) então chama o componente “*Tiny_DB1*” e armazena o valor na tag “*login*”, os dados do usuário, na parte do else, onde há a verificação se algum desses dados estiver preenchido de maneira errada e fecha a “*tela_login*” e abre a tela “*Lista_De_Compras*”.

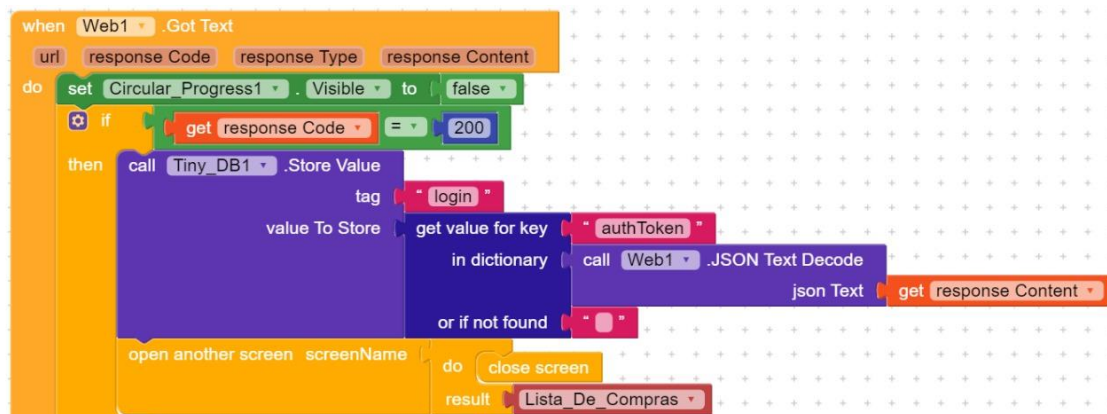


Figura 1 – Parte do bloco “Web1” Fonte: Material de aula

Ao inicializar a tela “*Lista_De_Compras*”, o bloco da Figura 2 é a primeira ação a acontecer, buscando os dados de login no banco de dados, verificando se a variável “*global token*” está vazia, se estiver será chamado o procedimento de “*logout*”³. Este procedimento está contido também dentro de “*BT_logout.Click*”⁴, que se clicado leva a tela de login, esta tela conta também com o botão “*BT_addnovoproduto*” que quando clicado chama o procedimento dentro dele contido, que é o de fechar a tela atual e abrir outra tela, neste caso a tela “*Screen1*”.

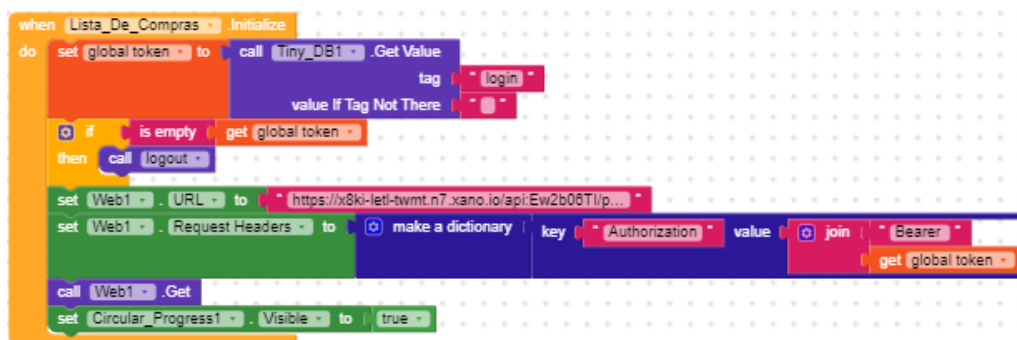


Figura 2 – Lista_de_Compras Initialize Fonte: Autoria Própria

No bloco da Figura 3, há a chamada do componente “*Web1*”, logo após ele ativa o “*Swipe Refresh*”, adiciona a variável “*global total*” na label

³ O procedimento *logout* não está neste texto, deve ser visualizado no Kodular

⁴ O bloco *BT_logout.Click* não está neste texto, deve ser visualizado no Kodular

“LB_precototal”, após esta ação ele chama o “List View” com a função “clear list”, para que quando o “Swipe Refresh” for acionado não aconteça a soma do valor que está contido na Label, pois este valor só será alterado quando algum produto novo for adicionado. Logo após o bloco “Web1” inicializar, verifica se os dados digitados pelo usuário são verdadeiros, então chama o bloco “Web1” e “List View”, dando início a formação da tabela.

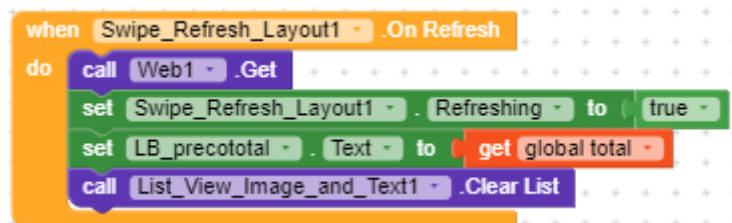


Figura 3 – Bloco Swipe_Refresh_Layout1 Fonte: Adaptado do material da aula

Na Figura 4 consta apenas uma parte do bloco para a formação da lista, podendo ser completamente visualizado no Kodular. Dentro do bloco da Figura 4, há a realização da soma de novos produtos adicionados ao total se houver. Neste bloco também consta a parte para verificação de expiração do login do usuário, se o login tiver expirado, o bloco irá redirecionar para a “tela de login”.

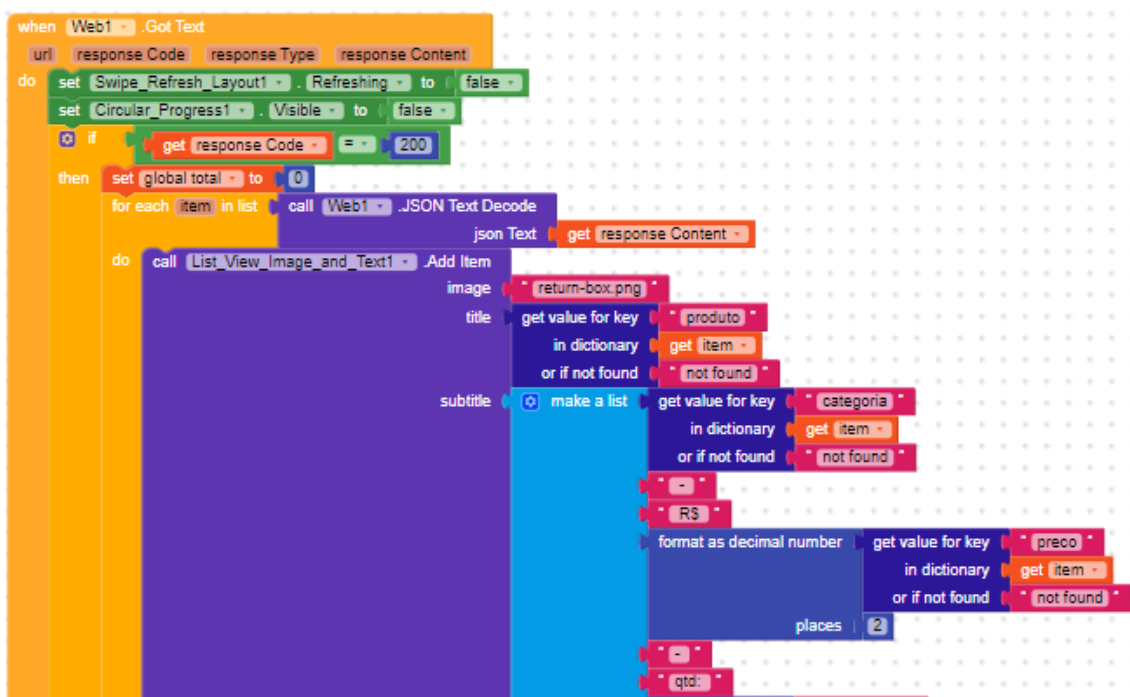


Figura 4 – Parte do Bloco Web 1 Fonte: Adaptado do material da aula

Quando a tela “*Screen1*” for inicializada, bloco “*Screen1*⁵” vai ser chamado, para que os dados do banco de dados possam ser pegos e verificar se a variável “*global token*” está ou não vazia, se estiver será chamado o procedimento “*logout*⁶”. O procedimento “*bt_salvar*⁷” que irá mudar a tela para a lista de compras, que será usado dentro do bloco “*Butt_salvar.Click*⁸” e o procedimento “*troca_tela*⁹”, que será usado quando o “*Butt_cancelar.Click*” for usado ou depois que um produto for adicionado dentro do bloco “*Web1*”. Quando o bloco “*Butt_salvar.Click*” for acionado, ele irá usar a Url gerada pelo XANO(2023) para que os dados sejam salvos nas variáveis “*produto*”, “*preco*”, “*quantidade*” e “*categoria*”, há uma verificação se estes campos não estão vazios para que seja atribuído um valor a chave, se algum destes campos estiver vazio haverá uma notificação que será gerada. O bloco “*Web1*” é chamado para que aconteça o armazenamento e busca de dados no banco de dados, verifica se os dados digitados pelo usuário são verdadeiros, com a visualização de uma notificação, e a mudança de tela na sequência. Há também um *else* para a verificação de erro na busca no banco de dados.

1.3 - XANO

Foi utilizado o XANO(2023), é uma plataforma *No Code* destinada a criação de *backends* para aplicativos *web* e *móveis*. Permite automatizar processos e integrar sistemas sem a necessidade de programação. Foi utilizado para armazenar os dados da lista de compras e de login dos usuários, como descrito na Figura 5. Todo o procedimento feito no XANO(2023) foi feito com base nas aulas disponibilizadas no Moodle da matéria.

⁵ O bloco *Screen 1* não está neste texto, deve ser visualizado no Kodular

⁶ O procedimento *call logout* não está neste texto, deve ser visualizado no Kodular

⁷ O procedimento *bt_salvar* não está neste texto, deve ser visualizado no Kodular

⁸ O bloco *Butt_salvar.Click* não está neste texto, deve ser visualizado no K

⁹ O procedimento *troca_tela* não está neste texto, deve ser visualizado no Kodular

The image shows two panels from the XANO interface. The left panel, titled 'ItemLista' with ID #263871, displays a table schema with the following fields: 'id' (integer), 'created_at' (timestamp), 'produto' (text), 'categoria' (enum), 'preco' (decimal), 'quantidade' (decimal), and 'totalitem' (decimal). The right panel, titled 'usuario' with ID #263416, displays a table schema with the following fields: 'id' (integer), 'created_at' (timestamp), 'email' (email), and 'senha' (password). Both panels show a dropdown arrow and an expanded view of the schema.

Table Name	ID	Fields
ItemLista	#263871	id: integer, created_at: timestamp, produto: text, categoria: enum, preco: decimal, quantidade: decimal, totalitem: decimal
usuario	#263416	id: integer, created_at: timestamp, email: email, senha: password

Figura 5 – Tabelas do XANO Fonte: Autoria Própria

O Link da Documentação da API gerada pela XANO é: <https://x8ki-letl-twmt.n7.xano.io/apidoc:eZ-duAJX/-/>

REFERÊNCIAS

1. Materiais de consulta e apoio disponíveis no Moodle da disciplina.
AS61A - Desenvolvimento Sem Código - N11 (2023_02)
<https://moodle.utfpr.edu.br/course/view.php?id=26209>
2. XANO - back-end sem código. Acessado em 20 de novembro de 2023.
<https://www.xano.com/>