

# Guia de Estudo - App Judô Conde Koma

Análise completa dos arquivos principais com comentários linha por linha

---

## Índice

1. package.json - Configurações e Dependências
  2. app.json - Metadados do Aplicativo
  3. App.tsx - Componente Raiz
  4. TabNavigator.tsx - Sistema de Navegação
- 

## 1. package.json

**Função:** Define dependências, scripts e configurações básicas do projeto

json

```
{  
  // Nome do projeto (usado internamente)  
  "name": "app-judo",  
  
  // Versão atual do aplicativo  
  "version": "1.0.0",  
  
  // ✅ PONTO DE ENTRADA - Arquivo que o Expo executa primeiro  
  // Este arquivo procura por App.tsx/App.js automaticamente  
  "main": "node_modules/expo/AppEntry.js",  
  
  // 🚀 SCRIPTS DE EXECUÇÃO  
  "scripts": {  
    "start": "expo start",      // Inicia servidor de desenvolvimento  
    "android": "expo start --android", // Abre direto no Android  
    "ios": "expo start --ios",    // Abre direto no iOS  
    "web": "expo start --web"     // Abre no navegador  
  },  
  
  // 📦 DEPENDÊNCIAS DE PRODUÇÃO (vão para o app final)  
  "dependencies": {  
    // 🎨 FONTES CUSTOMIZADAS  
    "@expo-google-fonts/inter": "^0.4.1", // Fonte Inter do Google  
    "@expo-google-fonts/lato": "^0.4.0",  // Fonte Lato do Google  
  
    // ⚙️ EXPO CORE  
    "@expo/metro-runtime": "~5.0.4",      // Runtime do bundler Metro  
    "@expo/vector-icons": "^14.1.0",      // Ícones vetoriais (Ionicons, etc.)  
  
    // 🧭 NAVEGAÇÃO REACT NAVIGATION  
    "@react-navigation/bottom-tabs": "^7.4.2", // Abas inferiores  
    "@react-navigation/native": "^7.1.14",    // Core da navegação  
    "@react-navigation/native-stack": "^7.3.21", // Stack navigation  
  
    // 🎨 ANIMAÇÕES  
    "animate.css": "^4.1.1",                // CSS animations (cuidado no mobile)  
  
    // 🏠 EXPO SDK  
    "expo": "~53.0.17",                    // Framework principal  
    "expo-app-loading": "^2.1.1",          // ⚠️ DEPRECATED - usar expo-splash-screen  
    "expo-font": "~13.3.2",                // Carregamento de fontes  
    "expo-linear-gradient": "~14.1.5",     // Gradientes lineares  
    "expo-localization": "^16.1.6",        // Localização/idioma  
    "expo-splash-screen": "~0.30.10",      // Tela de splash personalizada  
    "expo-status-bar": "~2.2.3",           // Controle da barra de status  
    "expo-updates": "~0.28.17",            // Updates over-the-air (OTA)
```

```
// 🌟 REACT
"react": "19.0.0",           // Biblioteca React
"react-dom": "19.0.0",       // React para web
"react-native": "0.79.5",     // React Native core

// 📺 ANIMAÇÕES REACT NATIVE
"react-native-animated": "^1.4.0", // Animações para RN
"react-native-linear-gradient": "^2.8.3", // ⚠️ DUPLICADO com expo-linear-gradient

// 📱 COMPONENTES NATIVOS
"react-native-safe-area-context": "5.4.0", // Área segura (notch, etc.)
"react-native-screens": "~4.11.1",        // Otimização de telas
"react-native-svg": "15.11.2",            // Suporte a SVG
"react-native-vector-icons": "^10.2.0",    // Ícones nativos
"react-native-web": "^0.20.0"              // React Native para web
},

// 🔧 DEPENDÊNCIAS DE DESENVOLVIMENTO (não vão para produção)
"devDependencies": {
  "@babel/core": "^7.25.2",    // Compilador JavaScript
  "@types/react": "~19.0.10", // Tipos TypeScript para React
  "typescript": "~5.8.3"       // Compilador TypeScript
},

// 🚫 Projeto privado (não será publicado no npm)
"private": true
}
```

### 📝 Observações importantes:

- **main:** Sempre deve ser `node_modules/expo/AppEntry.js` para projetos Expo
- **Dependencies:** Algumas duplicadas podem ser removidas
- **Scripts:** Comandos para executar o app em diferentes plataformas

## 2. app.json

**Função:** Configurações e metadados que o Expo usa para gerar o aplicativo

json

```

{
  "expo": {
    // 📱 INFORMAÇÕES BÁSICAS
    "name": "Judô Conde Koma", // Nome que aparece na tela inicial
    "slug": "judo-conde-koma", // ID único no Expo (URLs)
    "version": "1.0.0", // Versão visível ao usuário

    // 🖼️ INTERFACE DO USUÁRIO
    "orientation": "portrait", // Orientação: portrait/landscape/default
    "icon": "./assets/icons/app-icon.png", // Ícone principal (1024x1024px)
    "userInterfaceStyle": "automatic", // Tema: light/dark/automatic

    // 🌀 TELA DE SPLASH (carregamento)
    "splash": {
      "image": "./assets/splash/splash.png", // Imagem da splash
      "resizeMode": "contain", // Como redimensionar: contain/cover/native
      "backgroundColor": "#000000" // Cor de fundo
    },

    // 📁 ASSETS E RECURSOS
    "assetBundlePatterns": [ // Arquivos a incluir no bundle
      "**/*/*" // Todos os arquivos de todas as pastas
    ],

    // 🍏 CONFIGURAÇÕES iOS
    "ios": {
      "supportsTablet": true, // Suporte a iPad
      "bundleIdentifier": "com.rafaelvalverde.judocondekoma", // ID único na App Store
      "buildNumber": "1.0.0", // ⚠️ DEVERIA SER: "1"
      "infoPlist": { // Configurações do Info.plist
        "CFBundleAllowMixedLocalizations": true, // Múltiplos idiomas
        "ITSAppUsesNonExemptEncryption": false // Não usa criptografia
      }
    },

    // 📱 BARRA DE STATUS (Android)
    "androidStatusBar": {
      "backgroundColor": "#000000", // Cor de fundo
      "barStyle": "light-content" // Estilo: light-content/dark-content
    },

    // 🤖 CONFIGURAÇÕES ANDROID
    "android": {
      "adaptiveIcon": { // Ícone adaptativo (Android 8+)
        "foregroundImage": "./assets/icons/adaptive-icon.png", // Parte frontal
        "backgroundColor": "#FFFFFF" // Cor de fundo
      }
    }
  }
}

```

```

    },
    "package": "com.rafaelvalverde.judocondekoma", // ID único na Play Store
    "versionCode": 1 // Número interno da versão
  },

  // 🌐 CONFIGURAÇÕES WEB
  "web": {
    "favicon": "./assets/icons/favicon.png" // Ícone da aba do navegador
  },

  // ⚙️ CONFIGURAÇÕES EXTRAS
  "extra": {
    "eas": {
      "projectId": "626ae467-7eb4-483f-9352-5d4e7fbabece" // ID do projeto EAS
    }
  },

  // 🛠️ PLUGINS
  "plugins": [
    "expo-localization", // Suporte a localização
    "expo-font" // Carregamento de fontes
  ],

  // 👤 PROPRIETÁRIO
  "owner": "rafaelvalverde", // Username no Expo

  // 🔄 UPDATES OVER-THE-AIR
  "updates": {
    "enabled": true, // Habilita updates automáticos
    "fallbackToCacheTimeout": 0, // Tempo para fallback (0 = imediato)
    "url": "https://u.expo.dev/626ae467-7eb4-483f-9352-5d4e7fbabece" // URL do servidor
  },

  // 📺 VERSÃO DE RUNTIME
  "runtimeVersion": {
    "policy": "appVersion" // Usa versão do app como base
  }
}

```

### 📝 Observações importantes:

- **bundleIdentifier/package:** Devem ser únicos e consistentes
- **buildNumber:** Deve ser string numérica simples ("1", "2", etc.)
- **updates:** Permite atualizar o app sem passar pelas stores

### 3. App.tsx

**Função:** Componente raiz do React Native - primeiro a executar após inicialização

typescript

// 📖 IMPORTS - Bibliotecas necessárias

```
import { useFonts } from 'expo-font';           // Hook para carregar fontes
import * as SplashScreen from 'expo-splash-screen'; // Controle da splash screen
import { useCallback } from 'react';           // Hook de otimização do React
import { SafeAreaProvider } from 'react-native-safe-area-context'; // Área segura
import { NavigationContainer } from '@react-navigation/native'; // Container de navegação
import { StatusBar } from 'expo-status-bar';    // Controle da barra de status
import TabNavigator from './src/navigation/TabNavigator'; // Navegador principal
```

// 🚫 PREVINE AUTO-HIDE DA SPLASH SCREEN

```
// Splash screen só vai sumir quando dissermos explicitamente
SplashScreen.preventAutoHideAsync();
```

// 🏠 COMPONENTE PRINCIPAL

```
export default function App() {
```

// 📄 CARREGAMENTO DE FONTES

```
// useFonts retorna [fontsLoaded, fontError]
```

```
const [fontsLoaded] = useFonts({
  // Família Roboto (Material Design)
  'Roboto-Regular': require('./assets/fonts/Roboto-Regular.ttf'),
  'Roboto-Bold': require('./assets/fonts/Roboto-Bold.ttf'),
  'Roboto-Medium': require('./assets/fonts/Roboto-Medium.ttf'),
```

```
// Fonte customizada monospace
```

```
'BitcountPropSingle': require('./assets/fonts/BitcountPropSingle-Medium.ttf'),
```

```
// Família DM Sans (moderna e legível)
```

```
'DMSans-Bold': require('./assets/fonts/DMSans-Bold.ttf'),
'DMSans-Regular': require('./assets/fonts/DMSans-Regular.ttf'),
'DMSans-Medium': require('./assets/fonts/DMSans-Medium.ttf'),
'DMSans-ExtraBold': require('./assets/fonts/DMSans-ExtraBold.ttf'),
```

```
});
```

// 📱 CALLBACK PARA ESCONDER SPLASH SCREEN

```
// useCallback otimiza - só recria função se fontsLoaded mudar
```

```
const onLayoutRootView = useCallback(async () => {
  if (fontsLoaded) await SplashScreen.hideAsync(); // Esconde quando fontes carregaram
}, [fontsLoaded]); // Dependência: fontsLoaded
```

// ⌚ TELA DE LOADING

```
// Se fontes não carregaram, não renderiza nada (splash continua)
```

```
if (!fontsLoaded) return null;
```

// 🍌 RENDERIZAÇÃO PRINCIPAL

```
return (
```

```
// 📱 SafeAreaProvider - Gerencia áreas seguras (notch, etc.)
<SafeAreaProvider onLayout={onLayoutRootView}>
  {/* 📱 StatusBar - Controle da barra superior */}
  <StatusBar style="light" />

  {/* 🗺️ NavigationContainer - Container raiz da navegação */}
  <NavigationContainer>
    {/* 📄 TabNavigator - Sistema de abas principal */}
    <TabNavigator />
  </NavigationContainer>
</SafeAreaProvider>
);
}
```

### 🔄 Fluxo de execução:

1. **App carrega** → Inicia carregamento de fontes
2. **Fontes carregando** → Renderiza `null` (splash visível)
3. **Fontes prontas** → `onLayoutRootView` executa → Splash some → Interface aparece
4. **Interface renderiza** → SafeAreaProvider → NavigationContainer → TabNavigator

### 📝 Observações importantes:

- **useFonts**: Carrega fontes assincronamente
- **SplashScreen**: Controlado manualmente para melhor experiência
- **SafeAreaProvider**: Essencial para dispositivos com notch
- **NavigationContainer**: Obrigatório para React Navigation

## 4. TabNavigator.tsx

**Função:** Sistema principal de navegação com 5 abas e header personalizado

typescript



```
// 📄 IMPORTS
```

```
import React from 'react';
import { createBottomTabNavigator } from '@react-navigation/bottom-tabs'; // Criador de abas
import { Ionicons } from '@expo/vector-icons'; // Ícones do Expo
import { StyleSheet, Image, Pressable } from 'react-native'; // Componentes nativos
import { LinearGradient } from 'expo-linear-gradient'; // Gradientes
import { Texto } from '../components/Texto'; // Componente customizado
import { COLORS, FONT_SIZE, FONT_FAMILY } from '../theme/theme'; // Sistema de design
import * as Animatable from 'react-native-animatable'; // Animações
```

```
// 🏠 IMPORTAÇÃO DAS TELAS
```

```
import Home from '../screens/Home'; // Tela inicial
import Graduacao from '../screens/Graduacao'; // Graduações/faixas
import Historia from '../screens/Historia'; // História do judô
import Noticias from '../screens/Noticias'; // Notícias
import Perfil from '../screens/Perfil'; // Perfil do usuário
```

```
// 🗂️ CRIAÇÃO DO NAVEGADOR
```

```
const Tab = createBottomTabNavigator();
```

```
// 📝 MAPEAMENTO DE TÍTULOS
```

```
// Record<string, string> = objeto com chaves e valores string
```

```
const TITULOS: Record<string, string> = {
  "Início": "CONDE KOMA", // Tela inicial mostra nome da academia
  "Perfil": "Meu Perfil",
  "Notícias": "Notícias",
  "Graduação": "Graduação",
  "História": "História do Judô",
};
```

```
// 🎯 COMPONENTE PRINCIPAL
```

```
export default function TabNavigator() {
  return (
    <Tab.Navigator
      initialRouteName="Início" // Tela inicial
      screenOptions={({ route, navigation }) => ({ // Configurações para cada tela
```

```
// 🧠 HEADER PERSONALIZADO
```

```
header: () => (
  <LinearGradient
    colors={['#6D0F0F', '#A81412']} // Gradiente vermelho
    start={{ x: 0, y: 0 }} // Início: canto superior esquerdo
    end={{ x: 1, y: 1 }} // Fim: canto inferior direito
    style={styles.header} // Estilos do header
  >
    /* 🏠 LOGO - só aparece na tela inicial */
```

```
{route.name === 'Início' && (
  <Image
    style={styles.headerLogo}
    source={require('../assets/logo/logo.png')}
    resizeMode="contain" // Mantém proporção da imagem
  />
)}
```

```
{/* 📄 TÍTULO DINÂMICO */}
<Text style={styles.headerTitle}>
  {TITULOS[route.name] || route.name.toUpperCase()}
</Text>
</LinearGradient>
),
```

```
// 📱 ESTILO DA BARRA DE ABAS
tabBarStyle: {
  position: 'absolute', // Sobrepõe o conteúdo
  height: 65, // Altura fixa
  borderTopLeftRadius: 0, // Sem bordas arredondadas
  borderTopRightRadius: 0,
  backgroundColor: '#EDF2F4', // Cor de fundo clara
  elevation: 10, // Sombra no Android
  overflow: 'hidden', // Corta conteúdo que sair
},
```

```
// 🕒 BOTÃO PERSONALIZADO DE ABA
tabBarButton: (props) => (
  <Pressable
    android_ripple={{ color: '#8D99AE' }} // Efeito ripple (ondinha)
    onPress={props.onPress} // Função de toque
    style={props.style} // Estilos do botão
  >
    {props.children} // Conteúdo (ícone + texto)
  </Pressable>
),
```

```
// 📄 TEXTO DE CADA ABA
tabBarLabel: ({ focused }: { focused: boolean }) => (
  <Text style={focused ? styles.textFocused : styles.textUnfocused}>
    {route.name}
  </Text>
),
```

```
// 🍌 ÍCONES PERSONALIZADOS
tabBarIcon: ({ focused }: { focused: boolean }) => {
  // Tipo específico para TypeScript
}
```

```
let iconName: keyof typeof Ionicons.glyphMap = 'home';
```

```
// 🔍 SELEÇÃO DO ÍCONE BASEADO NA TELA
```

```
switch (route.name) {
```

```
  case 'Início':
```

```
    iconName = focused ? 'home' : 'home-outline';    // Casa
```

```
    break;
```

```
  case 'Graduação':
```

```
    iconName = focused ? 'medal' : 'medal-outline';    // Medalha
```

```
    break;
```

```
  case 'História':
```

```
    iconName = focused ? 'heart' : 'heart-outline';    // Coração
```

```
    break;
```

```
  case 'Notícias':
```

```
    iconName = focused ? 'newspaper' : 'newspaper-outline'; // Jornal
```

```
    break;
```

```
  case 'Perfil':
```

```
    iconName = focused ? 'person' : 'person-outline';    // Pessoa
```

```
    break;
```

```
}
```

```
// Estilo baseado no foco
```

```
const iconStyle = focused ? styles.iconFocused : styles.iconUnfocused;
```

```
// 🤖 ANIMAÇÃO DO ÍCONE ATIVO
```

```
return (
```

```
  <Animatable.View
```

```
    // Key única para resetar animação
```

```
    key={` ${focused ? 'focused' : 'unfocused'}-${route.name}`}
```

```
    // ✨ ANIMAÇÃO DE PULSO
```

```
    animation={focused ? {
```

```
      0: { transform: [{ scale: 1 }] },    // Tamanho normal
```

```
      0.5: { transform: [{ scale: 1.25 }] }, // 25% maior
```

```
      1: { transform: [{ scale: 1 }] },    // Volta ao normal
```

```
    } : undefined}
```

```
    iterationCount={focused ? 'infinite' : 1} // Infinita se ativo
```

```
    duration={1000}    // 1 segundo por ciclo
```

```
    useNativeDriver    // Otimização
```

```
  >
```

```
    <Ionicons name={iconName} style={iconStyle} />
```

```
  </Animatable.View>
```

```
);
```

```
},
```

```
)))
```

```
>
```

```
{/* 📱 DEFINIÇÃO DAS 5 TELAS */}
```

```
<Tab.Screen name="Início" component={Home} />  
<Tab.Screen name="Graduação" component={Graduacao} />  
<Tab.Screen name="História" component={Historia} />  
<Tab.Screen name="Notícias" component={Noticias} />  
<Tab.Screen name="Perfil" component={Perfil} />  
</Tab.Navigator>
```

```
);
```

```
}
```

```
// 🎨 ESTILOS
```

```
const styles = StyleSheet.create({
```

```
  // Header do topo
```

```
  header: {  
    flexDirection: 'row',      // Elementos lado a lado  
    alignItems: 'center',     // Centraliza verticalmente  
    justifyContent: 'flex-start', // Alinha à esquerda  
    paddingHorizontal: 20,     // Espaçamento lateral  
    height: 80,               // Altura fixa  
  },
```

```
  // Título do header
```

```
  headerTitle: {  
    fontSize: FONT_SIZE.xlarge, // Tamanho do seu tema  
    color: '#FFFFFF',           // Branco  
  },
```

```
  // Logo no header
```

```
  headerLogo: {  
    width: 60,                 // Largura fixa  
    height: 60,               // Altura fixa  
    margin: 10,               // Margem ao redor  
  },
```

```
  // Ícone ativo
```

```
  iconFocused: {  
    fontSize: 22,              // Maior  
    color: '#A81412',         // Vermelho da academia  
  },
```

```
  // Ícone inativo
```

```
  iconUnfocused: {  
    fontSize: 20,              // Menor  
    color: '#8D99AE',         // Cinza  
  },
```

```
  // Texto ativo
```

```
textFocused: {
  fontSize: 16,           // Maior
  color: '#A81412',      // Vermelho da academia
},

// Texto inativo
textUnfocused: {
  color: '#8D99AE',      // Cinza
  fontSize: 14,          // Menor
},
});
```

## Estrutura do TabNavigator:

1. **LinearGradient Header:** Header personalizado com gradiente vermelho
2. **Logo condicional:** Logo aparece apenas na tela "Início"
3. **5 Abas principais:** Início, Graduação, História, Notícias, Perfil
4. **Animações:** Ícones ativos têm animação de pulso infinita
5. **Estilos dinâmicos:** Cores e tamanhos mudam baseado no foco

## Observações importantes:

- **screenOptions:** Função que retorna configurações para cada tela
- **tabBarIcon:** Função que retorna o ícone baseado no estado focused
- **Animatable.View:** Biblioteca para animações fluidas
- **useNativeDriver:** Otimização que usa thread nativo para animações

## Fluxo Completo de Execução

1. package.json → Define entry point
2. AppEntry.js (Expo) → Procura App.tsx
3. App.tsx → Carrega fontes, configura providers
4. TabNavigator → Define navegação e telas
5. Telas individuais → Renderizam conteúdo específico

## Para converter em PDF:

1. **Online:** Copie este markdown para sites como [dillinger.io](https://dillinger.io) ou [stackedit.io](https://stackedit.io)
2. **VS Code:** Use extensão "Markdown PDF"
3. **Pandoc:** Use comando `pandoc arquivo.md -o arquivo.pdf`
4. **Typora:** Software que converte markdown para PDF

