

@hanafe/rn-release

`@hanafe/rn-release` is a CLI tool for managing releases of React Native projects. It helps with versioning, building, and deploying your app for both Android and iOS platforms.

This is a basic tool. More functionalities are to be added soon. For contribution contact me at miraahanafee@gmail.com

NOTE: A previous version of this same package named `rn-hanafe` is now deprecated

Features

- Update `package.json` version automatically.
- Update Android `build.gradle` `versionName` and `versionCode`.
- Update iOS `Info.plist` `CFBundleShortVersionString` and `CFBundleVersion`.
- Build release APKs for Android.
- Build archives for iOS.
- Create Git commits and tags for releases.
- Stream-install APKs to multiple connected Android devices.

Installation

Install the CLI globally using npm:

```
npm install -g @hanafe/rn-release
```

Usage

Run the CLI using the command:

```
@hanafe/rn-release <command> [options]
```

Commands

release

Prepare and build a new release for your React Native project.

```
@hanafe/rn-release release [options]
```

Options:

- `-v, --version <version>`: Specify the version type (`major`, `minor`, `patch`) or a specific version (e.g., `1.2.3`).
- `-p, --platform <platform>`: Specify the platform to build for (`android`, `ios`, `both`). Default is `both`.
- `-t, --tag`: Create a Git commit and tag for the release. Default is `true`.

Example:

```
@hanafe/rn-release release -v minor -p android
```

stream-install

Stream-install an APK to multiple connected Android devices via ADB.

```
@hanafe/rn-release stream-install
```

This command detects connected Android devices and installs the release APK (`android/app/build/outputs/apk/release/app-release.apk`) on selected devices.

init

Initialize project settings (functionality to be implemented).

```
@hanafe/rn-release init
```

Interactive Mode

If no options are provided for the

release

command, the CLI will prompt you for:

- Version type (`major`, `minor`, `patch`, or custom).
- Platform to build for (`android`, `ios`, or both).
- Whether to create a Git commit and tag.

Example Workflow

1. Navigate to the root of your React Native project.
2. Run the

release

command:

```
@hanafe/rn-release release
```

3. Follow the prompts to update the version, build the app, and create a Git tag.

Requirements

- Node.js >= 16.0.0
- For Android builds:
 - Android SDK and Gradle installed.
- For iOS builds:
 - Xcode and command-line tools installed.
 - **PlistBuddy** available (comes with macOS).

Troubleshooting

- Ensure you are in the root directory of your React Native project.
- Make sure

package.json

exists in the project root.

- For Android builds, ensure the **android/app/build.gradle** file exists.
- For iOS builds, ensure the **ios** directory contains a valid **.xcodeproj** or **.xcworkspace**.

License

This project is licensed under the MIT License.