Maximizing Next Generation Experiences w/ Mendix Pluggable Components

Danny Roest / Product Manager / Mendix
Mitya Kononchuk / Tech Lead / Mendix
Andries Smit / Team Lead & Developer / Mendix
Marco Laponder / Team Lead & Developer / Mendix
Agenda

1. Building widgets + Demo
2. Building actions + Demo
3. What's Next
Enablement through pluggability

Build powerful re-useable extensions based on popular technologies
Pluggable Widgets

- Successor of custom widgets
- Focused on Interface & Presentation
- ReactJS for Web and Hybrid mobile apps
- ReactNative for Native Mobile apps
- TypeScript & JavaScript support
- Shareable via (private) Appstore
Pluggable widgets - technology

**Widget as standard React Component**
- Use external dependencies
- Build with modern JavaScript or TypeScript

**Flux architecture**
No explicit data fetching needed
- Receive plain data through props
- Render UI component
- Dispatch actions

```javascript
// Widget

props.value
props.label
props.value.setValue()
props.action.execute()
```

Mendix client

```javascript
// Widget

props.value
props.label
props.value.setValue()
props.action.execute()
```
New Features

- Context derived automatically
- Lazy loaded
- External components usage
- Preview mode
More New Features

- Labeled
- Action parameter
- Text template parameter
- Expression parameters
Quick & Easy Widget Development Flow

Create project: `yo mx-widget`
- Web app or Native Mobile app
- JavaScript or Typescript
- Default or template

Run project: `npm start`
- Hot reload in browser
- Autocompletion for TypeScript
Extending an app
Demo time!

Widget for **Native Mobile** based on **JavaScript**

Widget for **web** based on **TypeScript**
Pluggable Nanoflow Actions

AKA JavaScript actions

- Build powerful device integrations
- Extend nanoflows with custom logic
- Code editor inside Mendix Studio Pro
- Expose as standard nanoflow activities
- Share via (private) AppStore
Demo time!
What’s next for Pluggable widgets?

- Support list of objects
- Contain other widgets
- Widget interactions
What’s next for Pluggable actions?

- Modern JavaScript & TypeScript Support
- External Dependencies
- More Parameter Types
Go make great extensions and enable other makers!
Thank you!
Questions?

How-To’s will be available with the release