



Front-end

# Sadržaj

- Atlas UI
- Gradivni elementi:
  - Layouts
  - Page, Pages templates
  - Widgets
- Rukovanje podacima:
  - Data view
  - Grid
  - Grid template
  - List view
  - Nasleđivanje
- Rukovanje vezama
- Linkovi
- Dodatno:
  - Layouts
  - Building blocks
  - Snippets
  - Stilizovanje

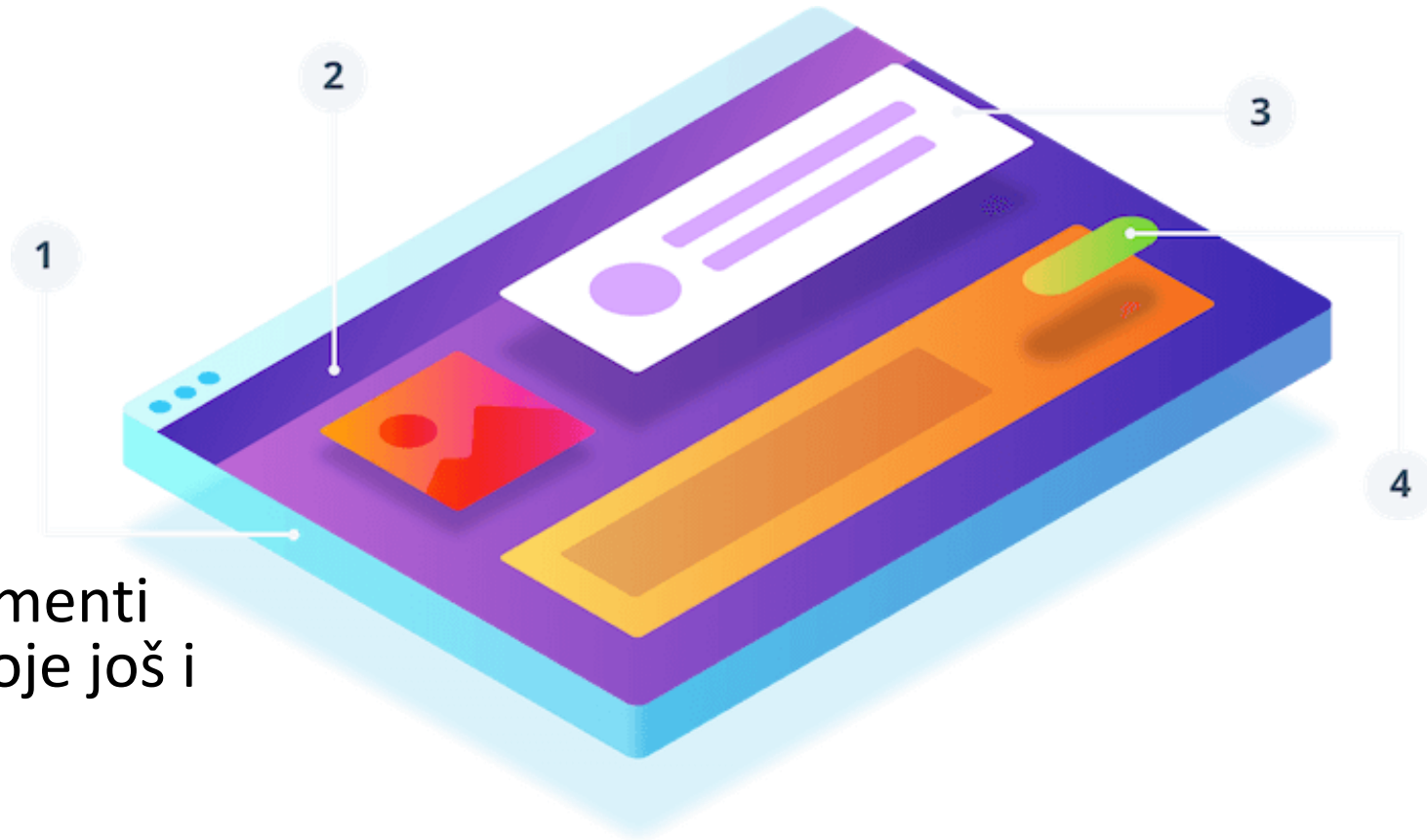
# Atlas UI

- Kako bi se ubrzao razvoj *front-end* dela aplikacija, Mendix dolazi uz [Atlas UI](#)
- *Atlas UI* predstavlja *design framework*, koji podrazumeva prethodno kreirane i **stilizovane** gradivne elemente, koji služe za izradu *front-end* dela aplikacije
- Pored postojećih gradivnih elemenata, u *framework* je moguće i proširiti: UI/UX dizajner kreira nove gradivne elemente, koje kasnije developer po potrebi može da iskoristi
- Tako proširen *framework* je moguće iskoristiti i za druge projekte, čime se olakšava pravljenje aplikacija sa konzistentnim izgledom

# Gradivni elementi

Postupak kreiranja stranice:

- Bira se struktura stranice (raspored za stranicu) (1. *Navigation Layout*)
- Kreira se sama stranica (2. *Page*), verovatno po šablonu (*Template*)
- Na stranicu se dodaju komponente (4. *Widgets*), često grupisane u gradivne elemente (3. *Building blocks*)
  - *Widget* -> *Dropdown*
  - *Building block* -> *Horizontal Form*



Navedeni elementi su gradivni elementi za *Mx Front-end*. Pored toga, postoje još i *Snippet*-i.

# Gradivni elementi - *Layouts*

- *Layout* elementi definišu strukturu za stranice (*Page*) - svaka stranica je bazirana na nekom od *Layout*-a. Sastoje se od sadržaja i mesta za dopunu (*placeholder*-a).
- Komponente (*widgets, building blocks*) postavljene u *Layout* će biti deo **svake** stranice bazirane na tom *Layout*-u.
- Primer: navigacioni meni i *footer* – njihov sadržaj ne mora da se kreira za svaku pojedinačnu stranicu!
- *Placeholder*-i definišu mesta koja bi stranica trebalo da popuni.
- Detaljnije o *Layouts* i o tome kako se prave videti u sekciji Dodatno

# Gradivni elementi - *Layouts*

- *Layout* može spadati u jedan od narednih 5 tipova:

Layout Type	Description
Responsive	Use this layout for pages that will work fine on all types of devices. The <a href="#">layout grid</a> and other widgets make it possible to create responsive pages.
Tablet specific	Use this layout for pages on a tablet if the responsive option is not sufficient (for example, if different use cases with different user interfaces have to be supported).
Phone specific	Use this layout for pages on a phone if the responsive option is not sufficient (for example, if different use cases with different user interfaces have to be supported).
Modal pop-up	Use this layout for pages that should appear as <a href="#">modal pop-up windows</a> .
Pop-up	Use this layout for pages that should appear as modeless pop-up windows.

# Gradivni elementi - *Pages*

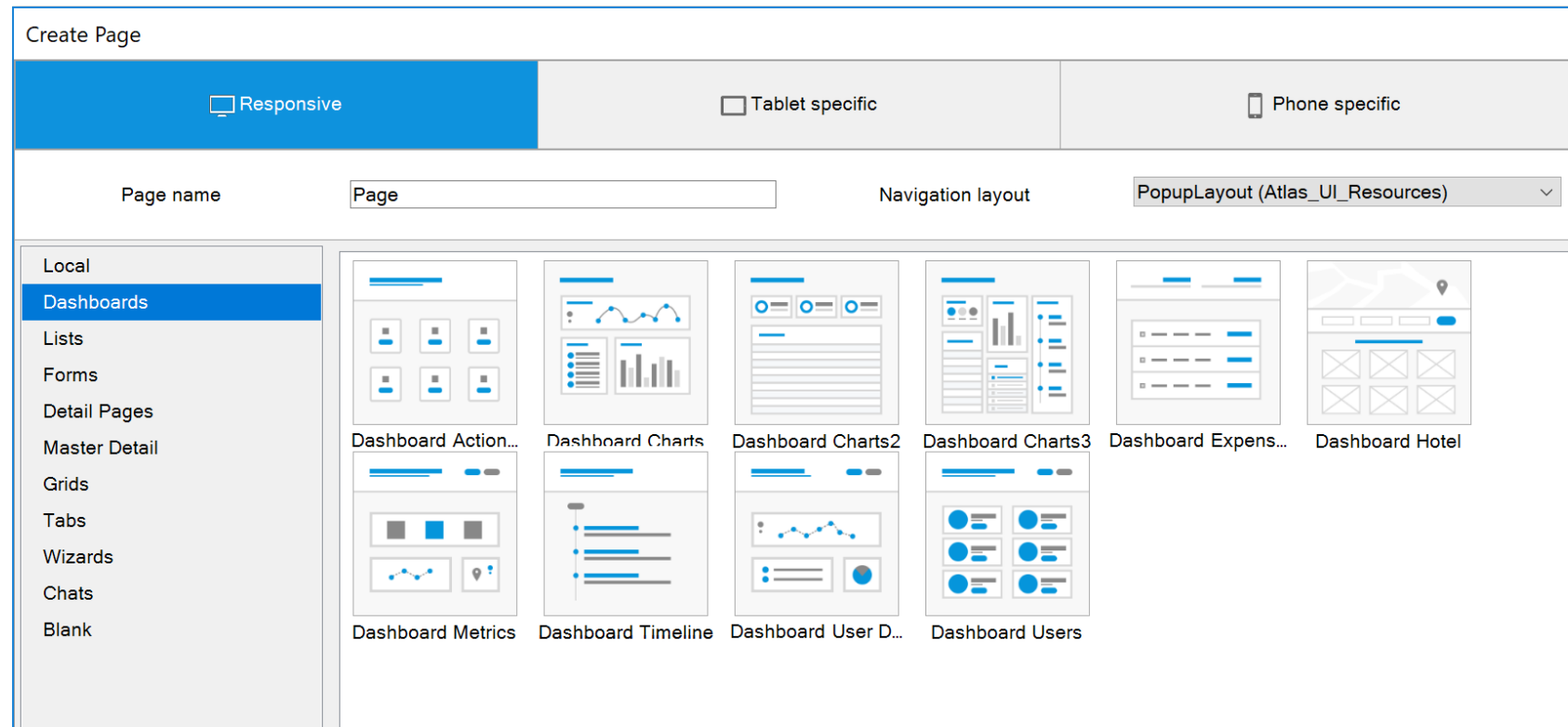
- Stranice definišu krajni korisnički interfejs Mendix aplikacije
- Bazirane su na nekom od *layout*-a, i komponentama popunjavaju "rupe"- mesta definisana pomoću *placeholder*-a
- Od osobina (*properties*) stranice bi trebalo spomenuti:
  - Naslov (*title*), koji se prilikom otvaranja stranice može menjati
  - URL koji odgovara stranici (što omogućava slanje linkova, i *bookmark*-ovanje):
    - Format je sledeći: `http://example.mendixcloud.com/p/url_stranice`
    - Ako stranica prima parametar, format je sledeći: `http://example.mendixcloud.com/p/url_stranice/{id}`, pri čemu id odgovara id internom id obeležju entiteta.  
Primer: `http://example.mendixcloud.com/p/order/3212449487634321`
    - Ako stranica ne definiše URL, on se dolaskom na tu stranicu neće menjati
- Postoje i *pop-up* osobine, vezana za rukovanje takvim stranicama (definiše se veličina, mogućnost njene promene i akcija zatvaranja)

# Gradivni elementi – *Page Templates*

- Kako bi se olakšao inicijalni napor kreiranja stranice, postoji više šablona koji mogu poslužiti za osnovu pravljenja stranice - *Page templates*.
- Izgled predefinisanih šablona, dokumentaciju i uputstvo možete pogledati [Atlas UI](#) stranici (templates sekcija). Predefinisani šabloni se nalaze u Atlas\_UI\_Resources modulu.
- Šablone je, takođe, moguće kreirati. Kreirani šabloni, koji su deo nekog od modula projekta, će se naći u odeljku *Local* na ekranu za kreiranje nove stranice.
- Kreirani šabloni imaju *Shown When* osobinu, kojima se određuje kada će biti ponuđeni korisniku modelera: kada se kreira nova stranica, kada se generiše stranica za izmenu, ili stranica za selekciju.

# Gradivni elementi – *Page Templates*

- Na slici ispod je prikazan prozor za kreiranje nove stranice. Unosi se ime, *layout*, i eventualno *page template*. Za kreiranje prazne stranice odabrati *blank template*.



# Gradivni elementi – *Widgets*

- [Widgets](#) (sekcija *Widgets/Web sekcija*) su osnovne gradivne komponente za izradu stranica, sadržaja na *layout*-ima, i *snippet*-a.
- Postoji više kategorija *widget* komponenti, i ne mogu sve biti korištene za izradu svih navedenih vrsta dokumenata (*Layouts, Page Templates...*):
  - *Menu widgets*, koji omogućavaju navigaciju
  - *Data widgets*, za prikaz i rukovanje podacima
  - *Layout widgets*, koji se koriste za definisanje strukture stranica
  - *Container widgets*, što mogu da sadrže druge komponente
  - *Input widgets*, koji omogućavaju izmenu atributa i asocijacija
  - *File widgets*, za rukovanje datotekama (i slikama)
  - *Button widgets*
  - *Report widgets*, koji omogućavaju kreiranje izveštaja i grafika nad (potencijalno agregiranim) podacima, koji ne moraju nužno biti skladišteni u bazi
  - *Chart widgets*, za kreiranje grafika
  - *Add-on widgets* – skinuti sa *App Store*-a, ili kreirani i uključeni u projekat

# Gradivni elementi – *Widgets*

- Osobinama *widget*-a možete upravljati kroz *Properties* prozor (desni klik na *widget* -> *Properties*)
- Opcije kojima je moguće upravljati zavise od tipa *widget*-a. One koje bi trebalo pomenuti jesu:
  - Opcija za vidljivost (*Visibility*). Može zavisiti od vrednosti nekog *boolean* ili enumeracijskog atributa
  - Opcije za *Input widget*-e
    - Omogućenje funkcionalnosti (*Editability*)
    - *Validation*, opcija koja zavisi od tipa *input* polja. *Front-end* validacija podataka
    - Data source – odabir atributa za koji se *input* polje veže
  - Opcije za akcije
    - *Events* skup atributa - šta se dešava na okidanje akcije (Save changes – Commit, Cancel changes – Rollback, i td.)

# Rukovanje podacima – *Data view*

- *Data view* je komponenta koja omogućava prikaz i izmenu jednog objekta u vidu forme. Pored forme, data view može da sadrži i *footer* sa kontrolama kao što je *Save* (ugnježdeni *data view* ne može imati *footer*)
- Pored forme sa podacima jednog objekta, omogućeno je prikazivanje i uvezanih objekata, kroz ugnježdene *data* komponente
- Izvor podatak za *data view* formu mogu biti:
  - Kontekst – parametar ili nadređeni *data view* (ovo ste koristili za *NewEdit* stranice)
  - *Microflow*
  - Oslušivani *widget* na istoj stranici (bitno je da *widget* vraća jedan element)
- Kontrole u komponentama koje rukuju kolekcijama, mogu automatski da proslede objekat kao parametar za *data view* na drugoj stranici (to se dešavalo kada klinkete na *Edit* dugme na *Overview* stranicama)

# Rukovanje podacima – *Data grid*

- *Data grid* je komponenta koja omogućava prikaz kolekcije elemenata u vidu tabele (nju ste koristili na *Overview* stranicama). Kontrole ove komponente omogućavaju operacije nad entitetima, pretragu, ali i izvoz podataka u CSV datoteke. Moguće je dodati i dodatne akcije, koje pozivaju *Microflow*-ove
- Komponenta podrazumevano omogućava sortiranje (kada je izvor podataka *XPath* ili *Database*) i paginaciju - obezbeđuje kontrole promenu redosleda i za menjanje stranice. Uz to, moguće je konfigurisati veličinu stranice (kroz broj redova u tabeli)
- Moguće je definisati vremenski period kroz koji se podaci osvežavaju
- Prikaz atributa se može konfigurisati: na primer, moguće je prikazati ime elementa enumeracije, ili sliku koja mu odgovara. Uz to, omogućeno je rukovanje širinom kolona

# Rukovanje podacima – *Data grid*

- Moguće opcije za selekciju elemenata iz *Data grid* komponente:

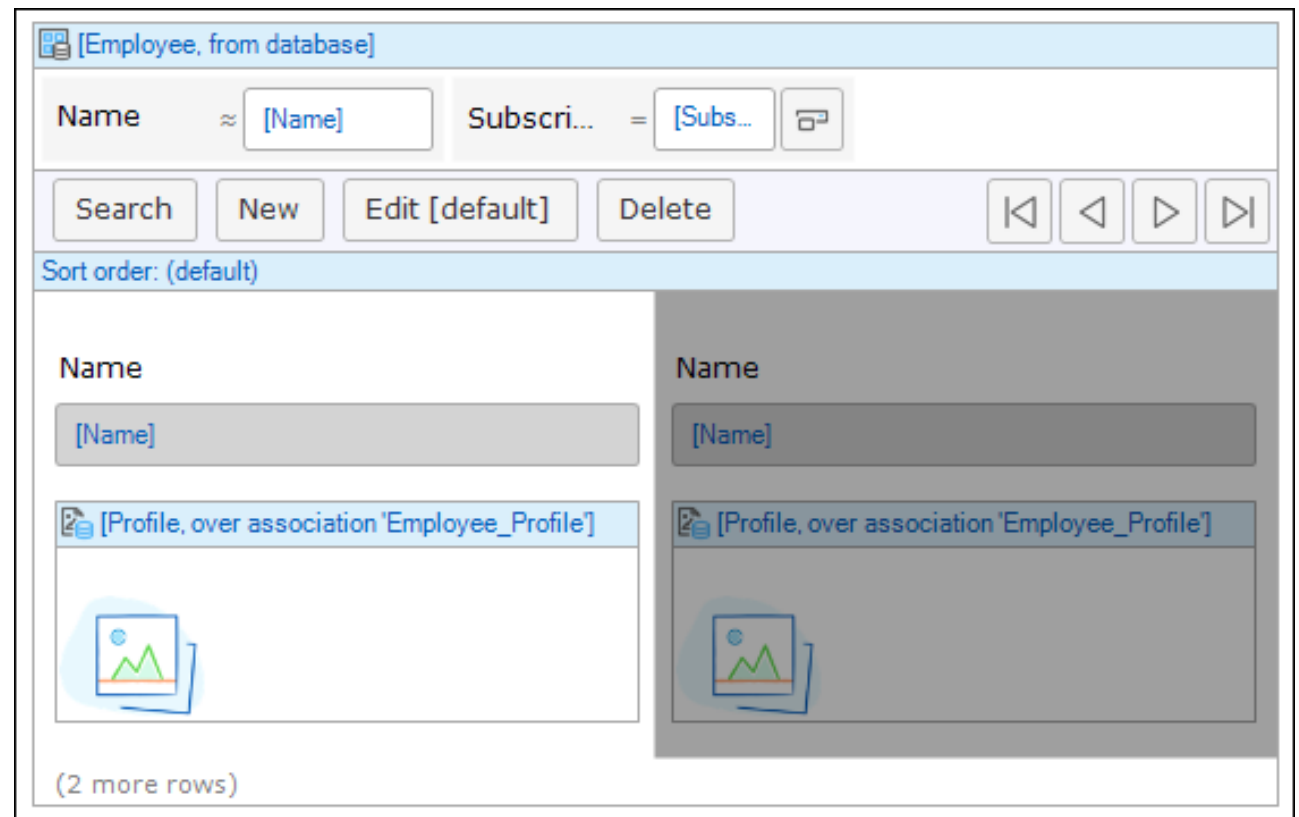
Value	Description
No selection	The user cannot select items.
Single selection	The user can select a single item by clicking on it. Clicking another item will make that item the selection.
Single selection and maintain	The user can select one item at a time by clicking on it. Users cannot deselect an item. By default the first item will be selected and removing a selected item will autoselect a subsequent item.
Multi-selection	The user can select multiple items by clicking the first one and holding the 'Ctrl' key while clicking on other items. Simply clicking an item will deselect all items and make the clicked item the selection.
Simple multi-selection	The user can select multiple items by simply clicking on them. This was the default multi-selection behavior prior to version 3.1.0.

- Ostale funkcionalnosti: izmena objekata kroz tabelu, **osvežavanje** **tabele u nekom vremenskom intervalu**, okidanje podrazumevane akcije, agregiranje podataka (omogućeno je u zavisnosti od izvora podataka)...

# Rukovanje podacima – *Data template*

- *Data template* komponente su vrlo slične *data grid* komponentama. Najveća razlika je u prikazu elemenata: dok se u *data grid*-u prikazuju u tabeli, kod *data template* komponente, one se prikazuju u šablonima uvezanim u *tile view* (koji se može podesiti da prikazuje određeni broj elemenata po redu)

- Dakle, može se prilagoditi izgled, za razliku od *Data Grid* komponente

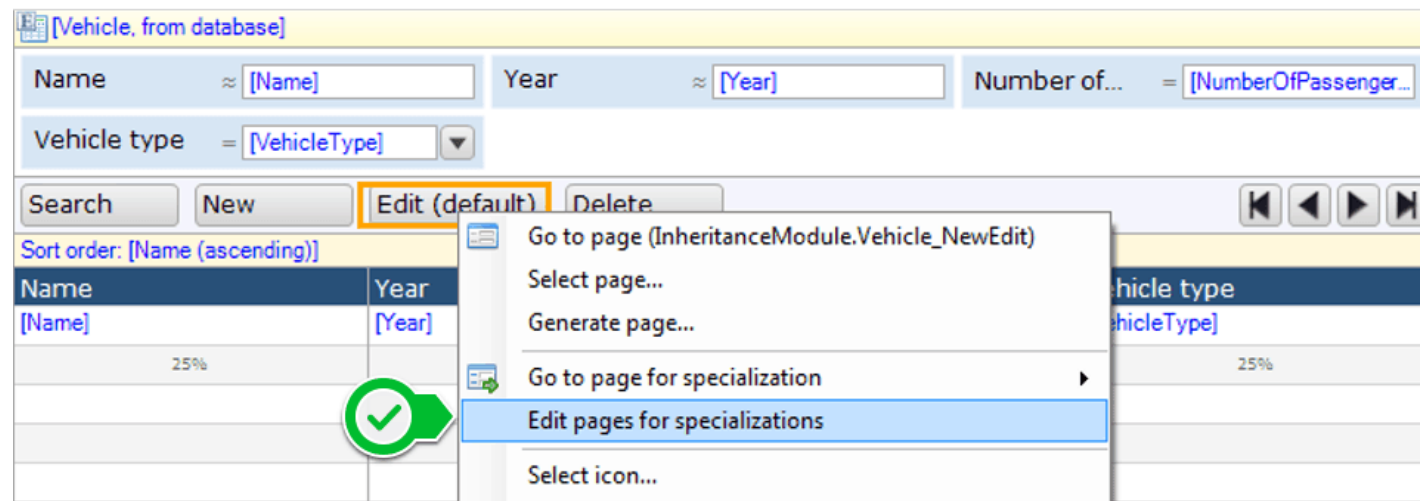


# Rukovanje podacima – *List view*

- I *list view* predstavlja komponentu za paginirani prikaz kolekcije elemenata zasnovanu na šablonima. Ovakvu komponentu ste sretali u mobilnim aplikacijama, npr. imenik
- Za razliku od *data template* komponente, raspored nije definisan *tile view*-om već je iskorišten vertikalni raspored – ali se on može menjati. Pored toga, ne postoje predefinisane kontrole i omogućena je isključivo kombinovana pretraga (po više obeležja odjednom)
- Kako su kontrole često potrebne, postoji [modul](#) na *app store*-u, koji omogućava razdvojenu pretragu po atributima i *bootstrap* paginaciju
- Navedene tri komponente za prikaz kolekcija (*data grid*, *data template* i *list view*) mogu imati sledeće izvore podatka:
  - *Database*
  - *XPath* (*XPath* ne radi u *offline* režimu rada u mobilnim aplikacijama)
  - *Microflow*
  - *Association* – kada u kontekstu postoji *data view*

# Rukovanje podacima – nasleđivanje

- Navedene komponente za prikaz kolekcija omogućavaju različito rukovanje nasleđivanjem:
  - *List view* omogućava različite šablone po specijalizaciji. Ako šablon za konkretnu specijalizaciju ne postoji, biće prikazan šablon za natklasu
  - *Data grid i data template* **ne** omogućavaju različit prikaz u zavisnosti od specijalizacije, ali omogućavaju različito ponašanje kontrola



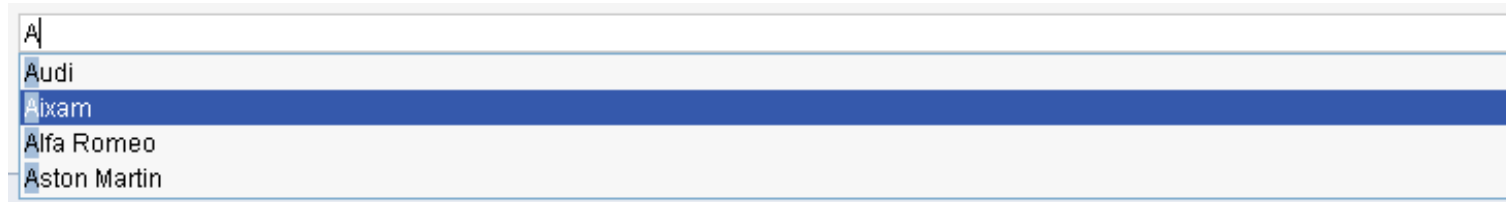
- O ovoj temi možete dodatno pročitati na sledećoj [stranici](#)

# Rukovanje vezama

- [Reference selector](#) je *widget* za rukovanje asocijacijama - omogućava da se vlasniku veze dodeli entitet za kog je vezan
- Omogućava selekciju u vidu *drop-down-a*, ili u vidu *pop-up* stranice (*select page*) u kojoj se nalazi *data widget* koji omogućava odabir – *drop-down* je preporučen za slučajeve gde je broj elemenata za odabir mali
- *Pop-up* stranicu je moguće i izgenerisati: desni klik na *reference selector* -> *generate select page*
- Za *reference selector* je potrebno odabrati atribut vezanog entiteta koji će biti prikazivan

# Rukovanje vezama

- [AutoComplete widget](#) je komponenta sa *app store*-a, pogodna je opcija kada je broj elemenata za odabir velik - tekstualna pretraga redukuje broj elemenata koji se jave u *drop-down* elementu. Obeležje po kojem se radi pretraga bi trebalo da bude jedinstveno. Posедуje opcije u vezi sa pretragom: broj prikazanih elemenata i minimalan broj karaktera koje je potrebno uneti da bi se prikazali rezultati.



# Rukovanje vezama

- [Reference set selector](#) predstavlja komponentu za odabir **skupa** vezanih entiteta (*many to many* veza). Entiteti koji jesu odabrani se nalaze u *data grid* komponenti. Kontrole *Add* i *Remove* rade dodavanje i uklanjanje veze između elemenata (a ne brisanje i dodavanje). *Select* kontrola u *pop-up* stranici će dodati odabrane elemente u listu vezanih elemenata

The screenshot displays a web application interface. In the background, there is a form with the following fields: 'Meet up id' (value: 11), 'Start date time' (placeholder: mm/dd/yyyy hh:mm AM/PM), 'End date time' (placeholder: mm/dd/yyyy hh:mm AM/PM), 'Max registrations' (value: 0), 'Topic' (value: MDE), and 'Location'. Below the form are 'Search', 'Add', and 'Remove' buttons. At the bottom, a 'Speaker ID' data grid is visible with two rows: SP1 (Name: Marija Maric) and SP3 (Name: Marko Milicic).

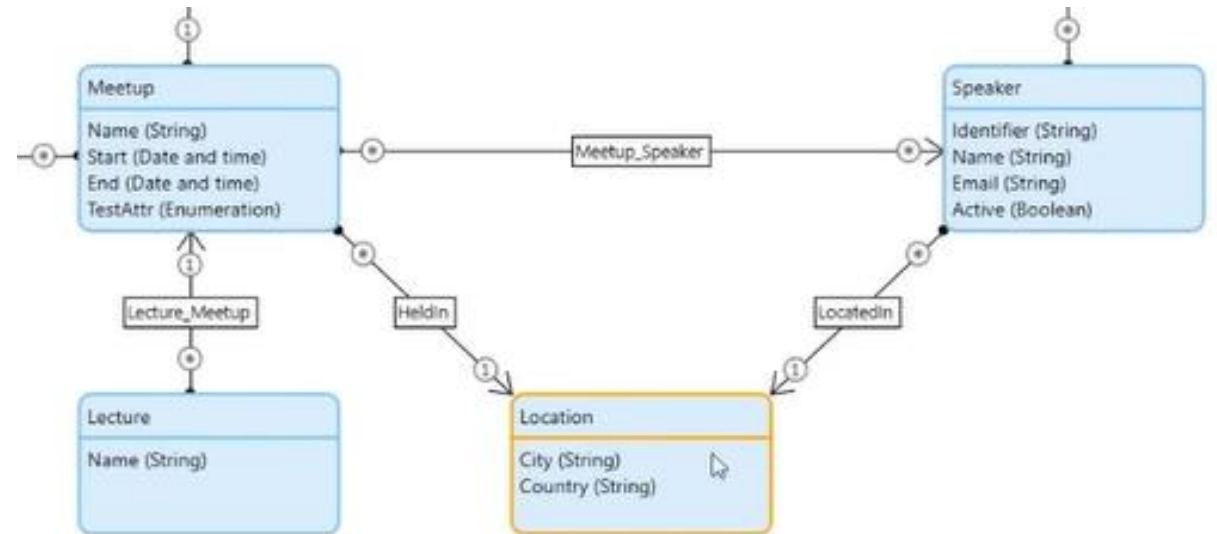
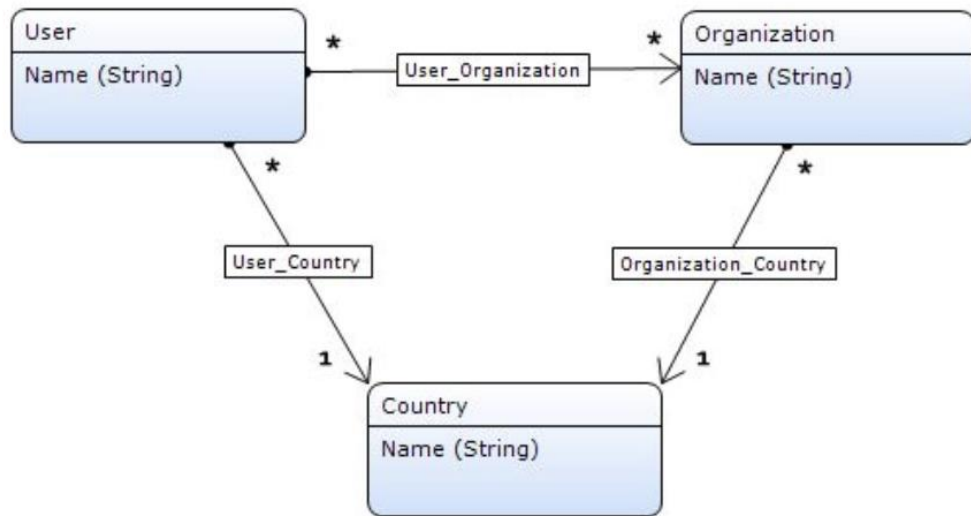
In the foreground, a 'Select Speaker' dialog box is open. It features a search bar with 'Search' and 'Select' buttons, and a pagination indicator showing '1 to 3 of 3'. The dialog contains a table with the following data:

Name	Email	Bio
Marija Maric	mm@gmail.com	
Marko Milicic	mm1@gmail.com	
Nikola Nikolc	nn@gmail.com	

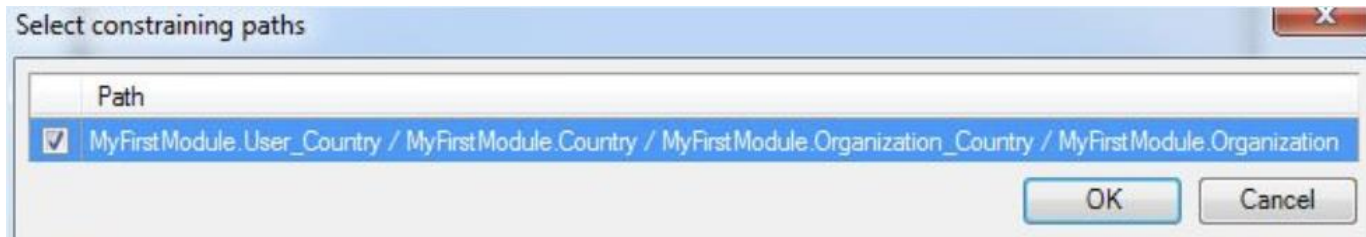
# Rukovanje vezama

- [Input reference set selector](#) radi isto što i *reference set selector*, samo što se selektovani elementi ne nalaze u data grid-u, već su samo pobrojani po nekom (jedinstvenom) atributu, razdvojeni zarezom
- Ono što je još bitno napomenuti jeste **restrikcija na elemente koje je moguće odabrati!** Primer: odabir robe koja jeste u magacinu. To je omogućeno kroz *XPath*.
- Uz restrikciju po *XPath* izrazu, moguće je odabrati i restrikciju po putanji između elemenata u domenskom modelu. Ovakva opcija je korisna kada odabir elemenata zavisi od nekog drugog elementa. Primer: odabir dobavljača iz neke konkretne države.

# Rukovanje vezama



<b>Country</b>	<input type="text" value="[User_Country/Country/Name]"/>
<b>Organization</b>	<input type="text" value="[User_Organization/Organization/Name]"/>



# Linkovi

- Omogućeno je kreiranje novih *widget*-a (zasnovanih na *React*-u), i njihovo uvođenje u projekte ili objavljivanje na *app store*. Uputstvo za to je dato na sledećem [linku](#)
- Prilikom izrade aplikacija, trebalo bi ispratiti najbolje [prakse](#)

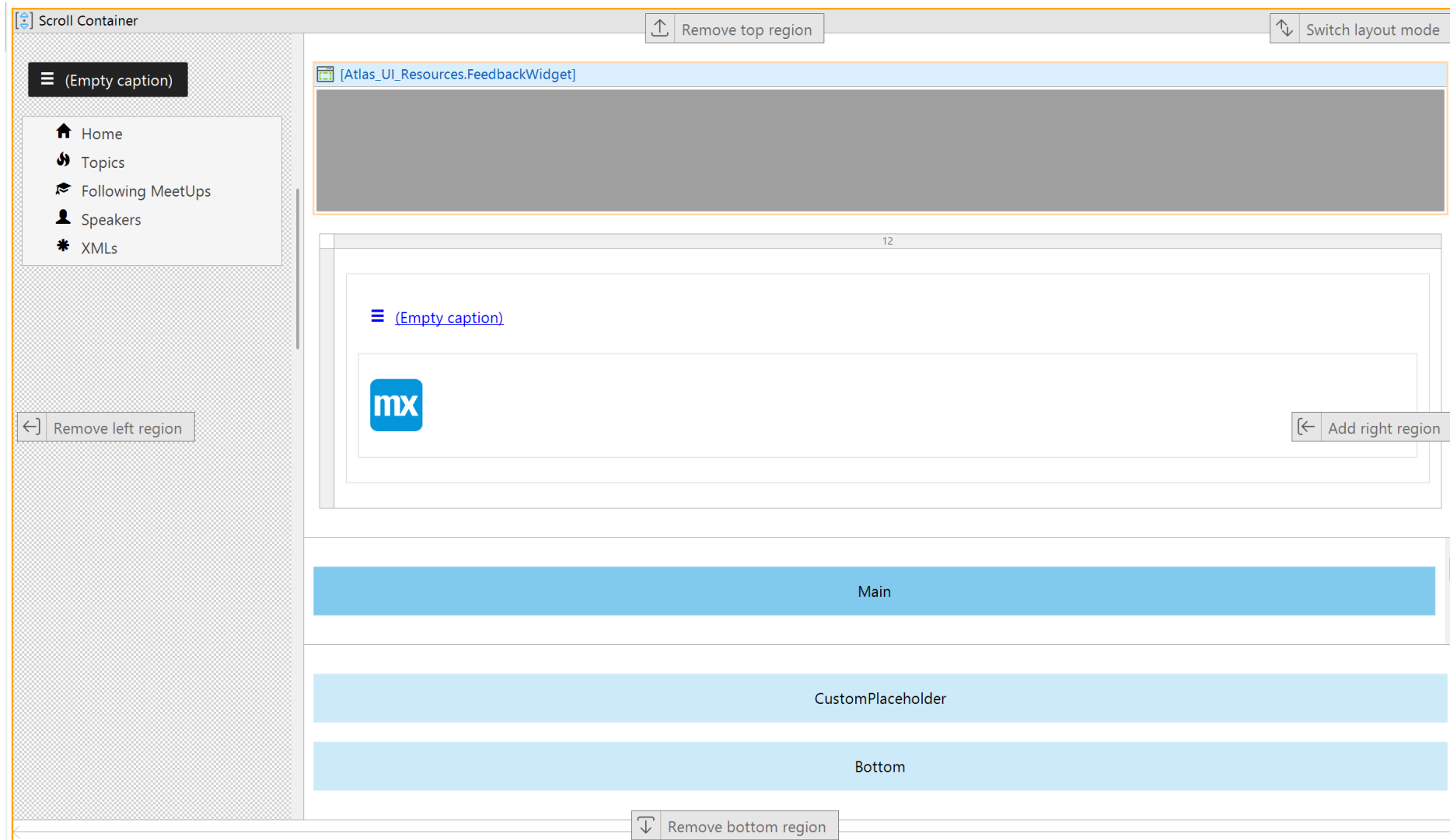


# Dodatno - Gradivni elementi - *Layouts*

- Kao što je rečeno, sadržaj u okviru [\*layout\*](#)-a će biti prisutan na svim stranicama koje su zasnovane na tom *layout*-u. *Mendix* optimizuje dovlačenje ovde definisanog sadržaja na klijenta.
- Sadržaj u okviru *layout*-a ne mogu biti komponente koje rukuju podacima (dakle *data widgets* ne mogu da se iskoriste u okviru *layout*-a, mada je moguće dodati *snippet* elemente, o kojima će biti reči nešto kasnije)
- Podrazumevano je osnovna komponenta u sadržana u *layout*-u [\*scroll container\*](#) (to ne mora biti tako), koji omogućava definisanje 5 regiona. *Scroll container* je *top-level* komponenta. Pored toga, u *layout* se dodaju navigacione komponente.

# Dodatno - Gradivni elementi - *Layouts*

- Primer *layout*-a: Stranica zasnovana na ovom layout-u bi trebalo da definiše sadržaj 3 *placeholder*-a (to se radi po imenu *placeholder*-a).

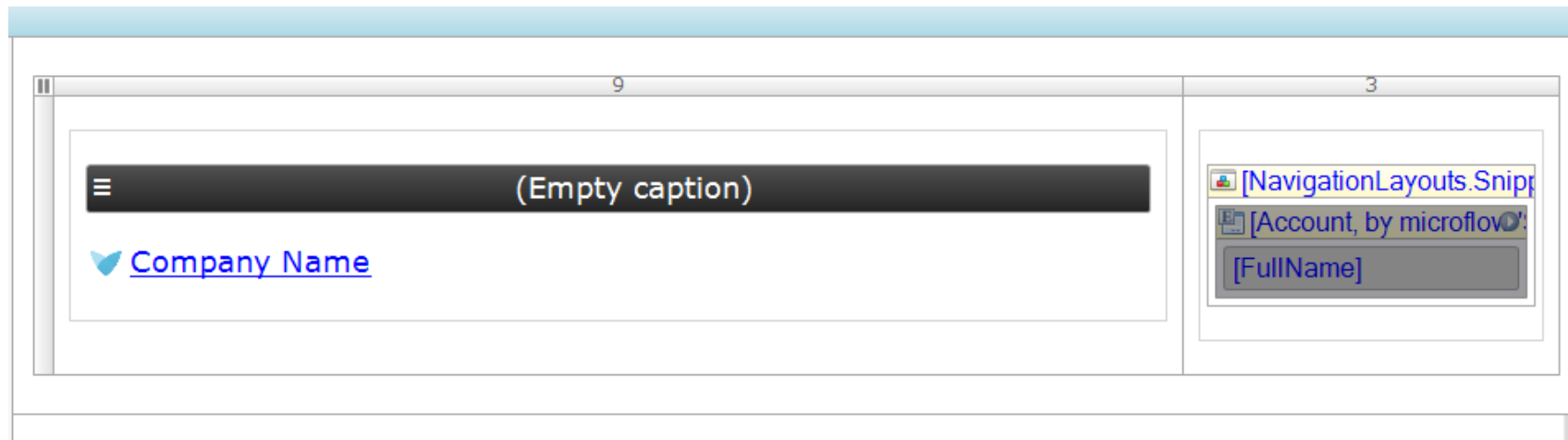


# Dodatno - Gradivni elementi – *Building Blocks*

- *Building blocks* predstavljaju prethodno stilizovane grupe *widget*-a, koji se često javljaju zajedno, kao što su elementi forme:
  - *Label, input, dropdown* su *widget*-i
  - Horizontalna i vertikalna forma su *building blocks*
- Izgled predefeinsanih *building block*-ova, dokumentaciju i uputstvo možete pogledati [Atlas UI](#) stranici (sekcija *Building Blocks*). I oni se nalaze u *Atlas\_UI\_Resources* modulu *Mendix* projekta
- Možete kreirati svoje *building block*-ove, kada uočite da se neke komponente često zajedno grupišu
- Uloga *building block*-ova je da ubrzaju dizajniranje, a ne da pruže funkcionalnost.

# Dodatno - Gradivni elementi – *Snippets*

- *Snippet-i* predstavljaju ponovno iskoristive delove interfejsa, koji mogu obuhvatati i funkcionalnost. Ugrađuju se u stranice i *layout-e*. Menjanjem *snippet-a*, promene se reflektuju na stranice u koje je ugrađen - olakšavano održavanje stranica
- Mogu biti vezane za neki od entiteta, kako bi se prikazali podaci. Tada se *snippet* može da se ugradi u neku *data* komponentu (obrađenim na narednim stranicama), i preuzme njene podatke iz konteksta



# Dodatno - Stilizovanje

- Kao što je već pomenuto, *Atlas UI framework* uključuje komponente koje su već stilizovane. Sam *framework* je zasnovan na *responsive design* principu, i u osnovi se nalazi *Bootstrap*
- *Framework* je pisan u [Sass](#)-u (u *scss* sintaksi), proširenju CSS-a, tako da je za izmene potreban transpajler. Preporuka je da iskoristite [Mendix Calypso](#)
- Izmene vrednosti *Sass* promenljivih je moguće izvršiti menjanjem *\_custom-variables.scss* iz `[Mxprojekat]theme/styles/web/sass/app` direktorijuma. U istom direktorijumu se nalazi i *custom.scss*, koji bi trebalo da posluži za nadogradnju. Posle izmena, transpajler mora da pretvori *sass* u *css*



# Dodatno - Stilizovanje

- Moguće je dodati i [dodatne css datoteke](#)
- Jednostavnije izmene su moguće kroz *Web Modeler* [Theme Customizer](#)
- U *Properties* prozoru za grafičke elemente (desni klik na element – *Properties*) postoji zaseban *Appearance* tab za opcije koje se tiču dizajna
- Moguće je definisati kako se elementima [dinamički dodeljuju klase](#)
- Redosled kojim se primenjuju CSS stilovi je sledeći:
  - Stil definisan temom
  - Klase definisane nad elementom
  - Stil definisan nad elementom



Hvala na pažnji!

