

# Oracle Forms alkalmazások modernizálása Java technológiával



Gere István  
Kereskedelmi tanácsadó  
Webváltó

**Bemutató**

**Forms modernizáció általános kérdései**

**Projekt tapasztalatok**

**A mi megoldásunk**

**DevBench program csomag**

# Az előadás tartalma



WEBVÁLTÓ

# Webváltó bemutatkozás: Oktatás

## Oracle University hivatalos oktatóközpont

- Tantermi képzések
  - Meghirdetett tanfolyamok
  - Privát, testreszabott tanfolyamok
- Live Virtual Class
- Digitális képzések
  - Training On Demand
  - Learning Subscription
  - Mentoring

## Saját fejlesztésű tanfolyamok



ORACLE

Approved  
Education Center



WEBVÁLTÓ

# Webváltó bemutatkozás: Szoftver infrastruktúra támogatás

- Üzemeltetés (Ops)
- Fejlesztés támogatás (DevOps)
- Konzultáció
- Cloud Migration Jumpstart Pack



# Webváltó bemutatkozás: Szoftverfejlesztés

- Egyedi szoftverfejlesztés
- On-site support



## Elmúlt hét évünk statisztikája

195

Fejlesztési  
projekt

=

151 553

Fejlesztéssel  
töltött óra

=

18 944

Munkanap

=

76

Év



WEBVÁLTÓ

# Fejlesztési referencia példák



- realCity
- MVH
- Banki referenciák  
Csaláskezelő rendszer



WEBVÁLTÓ



# Forms fejlesztés

*The biggest advantage of Forms is that it has about 20 years of rapid application development history.*

*As a result, productivity can be very high - in my case I always felt I could develop and deploy in 1/2 to 1/5 the time of any other language I wanted to use.*

# Miért érdemes modernizálni?

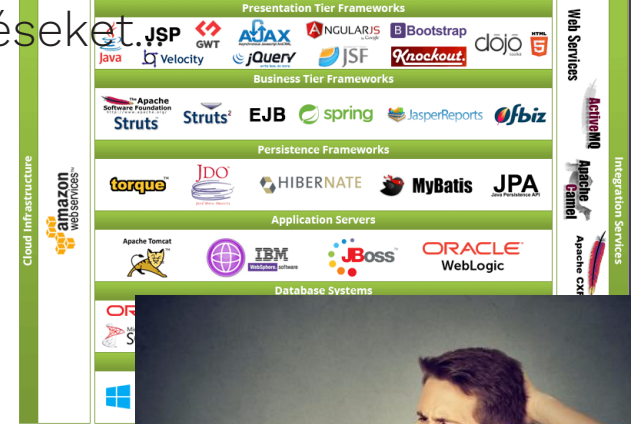
- Nehéz integrálni új technológiákhoz őket
- Egyre nehezebb Forms fejlesztőket találni
- Regression problémák
- Verziókezelés nem vagy csak nehezen megoldható
- A felhasználók igényeire nehezebben / lassabban lehet reagálni
- Megszűnik a keretrendszer támogatása

Release	GA Date	Premier Support Ends	Extended Support Ends	Sustaining Support Ends
Forms and Reports 11.1.2.x	Oct 2011	Dec 2016	Dec 2018	Indefinite
Forms and Reports 12.2.x	Oct 2015	Oct 2020	Oct 2023	Indefinite



# A modernizációs projektek problémái

- A Legacy rendszer fejlesztőinek nincs rá idejük
  - Jelentős paradigma váltást igényel
  - 100+ technológia amiből választani kell (helyesen)
  - Közben tartásuk is karban a meglévő fejlesztéseket
- Fejlesztői csapat bővítése az új technológiát ismerő emberekkel
  - Üzleti folyamatok ismeretének hiánya
  - Technológiai "kapzsiság,,
- Változások követésének problémája
  - Nem lehet megállítani az életet egy üzleti területen, azért mert épp szoftver átalakítás folyik
  - Tiger team effektus



# Mi lehet a megoldás?

- Dobozos termék

Egyre többen döntenek úgy, hogy inkább dobozos termékhez igazítják az üzleti folyamatokat  
De nem mindenre létezik dobozos termék

- Outsourcing

Sok esetben a fejlesztések kiszervezése megoldásnak tűnik

Viszont, üzleti tudás hiányában a beszállítók hatékonysága kisebb lehet mint a házon belüli fejlesztéseké

- **A házon belüli fejlesztés folytatása a nehézségek ellenére**



WEBVÁLTÓ

# A mi megoldásunk Konceptió

- Az üzleti tudással rendelkező Forms fejlesztők együtt dolgoznak a Java fejlesztőkkel.  
Egy közösen használt fejlesztőeszközzel.



- Az elkészülő alkalmazás a már bevált és a Java fejlesztők által ismert open source megoldásokra épül



open source



WEBVÁLTÓ

# Projekt tapasztalatok

## Ügyfél:

Amikor megszületik a döntés a modernizációról, akkor a következő kérdés mindig az, hogy de hogyan?

A hogyan?-t sokféleképpen fel lehet tenni?

- Hogyan lesz nekem modern alkalmazásom ami még talán többet is tud mint a mostani?
- Hogyan fogok olyan eszközöket/csapatokat találni, amelyek a segítségemre lesznek?
- Hogyan fogjak neki?
- Hogyan képzeljem el a modern alkalmazásomat?
- Hogyan...?



# Projekt tapasztalatok

**Amikor egy modernizációs feladatban találjuk magunkat, akkor nekünk is kérdéseink vannak:**

- Mi a cél?
- Hány modul van?
- Modulok kategorizálása: Lekérdező, adatrögzítő, riportáló, egyéb alapján.
- Hány felület van? Hány Report van?
- Az üzleti logika hol van? Adatbázis oldalon? Forms oldalon? És mi lenne a szimpatikus?
- A rendszernek létezik-e bármilyen dokumentációja? A fejlesztője elérhető-e, tudunk-e vele beszélni?



WEBVÁLTÓ

# Projekt tapasztalatok

**Amikor egy modernizációs feladatban találjuk magunkat, akkor nekünk is kérdéseink vannak:**

- Felhasználókkal tudunk-e egyeztetni, illetve velük közösen az igényeket kitalálni?
- Hány felhasználó van? Milyen jellegű napi feladataik vannak?
- Milyen külső rendszerintegrációk vannak?
- Hány felület van? Hány Report van?
- Az üzleti logika hol van? Adatbázis oldalon? Forms oldalon? És mi lenne a szimpatikus?
- Authentication and authorization



WEBVÁLTÓ

# Projekt tapasztalatok

**Amikor egy modernizációs feladatban találjuk magunkat, akkor nekünk is kérdéseink vannak:**

- Adatbázis kérdése: Kódtisztítás, normalizálás, teljes újra tervezés, vagy valami más?
- Igény-e, hogy a jelenlegi és az új rendszer (moduljaiban részben-egészben) működnek-e egymás mellett? Továbbá mi az átállás folyamata?
- Kik állnak a rendelkezésre a modernizáció során és a modernizáció befejeztével?
- A Projekt végeztével hogyan történik a továbbfejlesztés/karbantartás?



# Projekt tapasztalatok

## További kérdések:

- Fejlesztőeszközök
- Módszertanok
- Tesztelés
- Verziókezelés
- Pilot
- Oktatások
- Külső eszközök
- Projektek
- Emberek
- Pénz
- Design
- Dokumentáció
- Security
- Microservices
- Queue



WEBVÁLTÓ



# A mi megoldásunk

## Saját fejlesztésű modernizációt támogató programcsomag

### ■ DevBench

- UIBuilder
- AppDesigner
- Forms plugin
- További plugin-ok



WEBVÁLTÓ

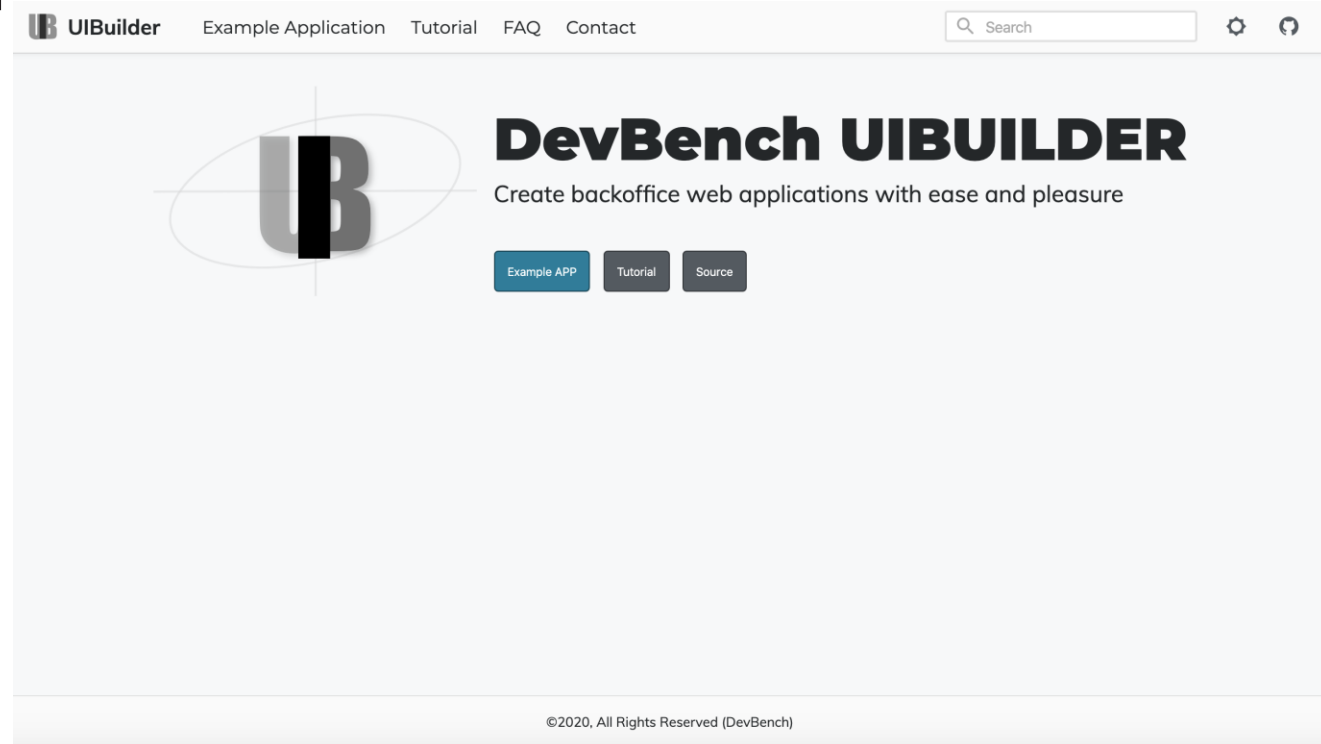
# UIBuilder

- Open source technológiákra épülő "keretrendszer"
- Cél: minimális kóddal működő alkalmazásokat készíteni



# UIBuilder

<https://uibuilder.devbench.io>



The screenshot shows the website for UIBuilder. At the top left, there is a navigation menu with the following items: **UIBuilder**, [Example Application](#), [Tutorial](#), [FAQ](#), and [Contact](#). To the right of the menu is a search bar with a magnifying glass icon and the text "Search". Further right are a gear icon for settings and a refresh icon. The main content area features a large logo on the left consisting of a grey "U" and a black "B" inside a white circle with a crosshair. To the right of the logo, the text reads **DevBench UIBUILDER** in a large, bold font, followed by the tagline "Create backoffice web applications with ease and pleasure" in a smaller font. Below the tagline are three buttons: "Example APP" (blue), "Tutorial" (grey), and "Source" (grey). At the bottom of the page, there is a copyright notice: "©2020, All Rights Reserved (DevBench)".

# UIBuilder

## Példa alkalmazás: <https://example.devbench.io>

The screenshot displays the UIBuilder application interface. At the top, there is a navigation bar with a home icon, a menu, and several tabs: Events / binding, Static resource, Flow Parameters, Page Scope, Window, Grid, TreeGrid, Combobox, Form, and a dropdown arrow. A 'Login' button is located on the far right of the navigation bar.

### Inline editable UIBuilder Grid with Spring-Data Repository

- The bean validation of the `todo` field validates if the field contains at least 3 characters. the default selection after the page load.
- The page `on-after-load` event handler is necessary only if using the grid without a crud-panel. If using inside a crud-panel, the datasource name is autodetected.
- Currently (1.3.0+) the `<uibuilder-item-editable>` cannot contain data-source dependency components, like a `uibuilder-grid`, or a `uibuilder-combobox` because of the need of the backend instantiation and the `item` property name collision. However, the pure vaadin components can work (for example `vaadin-combo-box`) if the `items=""` binding is not `item.` prefixed.

Below the text, there is a control bar with two buttons: 'Set all item to edit mode' (blue) and 'Save all edited' (red).

Task	Deadline
<input type="checkbox"/> Buy some chocolate	2021-04-28
<input checked="" type="checkbox"/> <input type="checkbox"/> Write more tests	6/28/2020 <input type="button" value="🗑"/>
<input type="checkbox"/> Brew a cup of coffee	2020-05-01

In the bottom right corner, there is a 'UIBuilder versions' panel with a list of snapshots: 2.0.1, 1.3.0, and 1.2.1.

# UIBuilder

Példa alkalmazás: <https://example.devbench.io>

The screenshot displays the UIBuilder application interface. At the top, there is a search bar with a '+' icon and a 'Search' button. Below it, a filter section shows 'Deadline' in a dropdown menu, followed by a list of dates: 22/04/2020, 02/04/2020, 08/04/2020, and 23/04/2020. A 'Task' list on the left contains: 'Buy some chocolate', 'Write more tests', and 'Brew a cup of coffee'. The main area shows a 'Todo item' editor for 'Brew a cup of coffee' with a deadline of '01/05/2020'. Below this is a table of reminders:

Text	Alert on day	Priority
Another reminder it...	2020-04-28	50
Last reminder	4/28/2020	75

A calendar widget is open, showing 'April 2020' with the 28th highlighted. A 'Reminder' form is also visible, with fields for 'Text' (Another reminder item), 'Alert day' (28/04/2020), and 'Priority' (50). At the bottom right, there is a 'UIBuilder versions' panel showing a 'Snapshot' with versions 2.0.1, 1.3.0, and 1.2.1. The bottom of the interface has 'Cancel', 'Reset', and 'Save' buttons.



WEBVÁLTÓ

# UIBuilder

## Példa alkalmazás: <https://example.devbench.io>

Home Menu Events / binding Static resource Flow Parameters Page Scope Window Grid TreeGrid Combobox Form ↓ Login

### Calendar

Green  Blue

< > today

27 Apr – 3 May 2020


month week day list

	Mon 27/04	Tue 28/04	Wed 29/04	Thu 30/04	Fri 01/05	Sat 02/05	Sun 03/05
all-day				Event 2 Event 2 extra property jn7h2			
08:00							
09:00		09:00 - 10:00 Green event					
10:00			10:00 - 12:30 Event 1 Event 1 extra property d1xs5n		10:00 - 12:00 Event 3 Event 3 extra property jmidq		
11:00							
12:00							

calendar.html CalendarControllerBean.java CalendarEntryRepository.java EntryWithDescription.java EntryWithDescriptionBuilder.java

```
<div style="display: flex; height: calc(100vh - 350px)">
<div style="padding-right: 1em;">
<vaadin-checkbox id="green" on-change="calendarControllerBean::filterChanged"
style="display: block;">
Green
</vaadin-checkbox>
<vaadin-checkbox id="blue" on-change="calendarControllerBean::filterChanged"
style="display: block;">
```

UIBuilder versions copy  
Snapshot  
2.0.1  
1.3.0  
1.2.1



WEBVÁLTÓ

# UIBuilder

## Tutorial: <https://uibuilder.devbench.io/tutorial-v1/>

UIBuilder Example Application **Tutorial** FAQ Contact

# Frontend

[UIBuilder](#) » [Tutorial \(V1\)](#) » [Part III](#) » Frontend

**Frontend**

Prolog

Part I >

Part II >

Part III ▾

Part III - Introduction

**Frontend**

Part IV >

Part V >

Part VI >

Part VII >

To actually manage the entities from a page UIBuilder provides a couple of options, first let's take a look at the most basic of them.

## Crud panel (without controller bean)

The first UIBuilder component to try out is the `crud-panel`. This composite component includes other components and functionalities to ease the CRUD UI creation process.

Open the `car.html` and remove the `<vaadin-date-picker></vaadin-date-picker>` from the `uibuilder-page`, it won't be needed anymore.

To start using the `crud-panel`, import it by adding the following line to the first row of your html file:

```
1 <link rel="import" href="frontend://bower_components/crud-panel/crud-panel.html">
```

Next put a `<crud-panel>` tag pair inside the `<uibuilder-page>` tag pair, and set the `orientation` property of the component to `horizontal`:

### What's on this Page

. Crud panel (without controller bean)

# UIBuilder

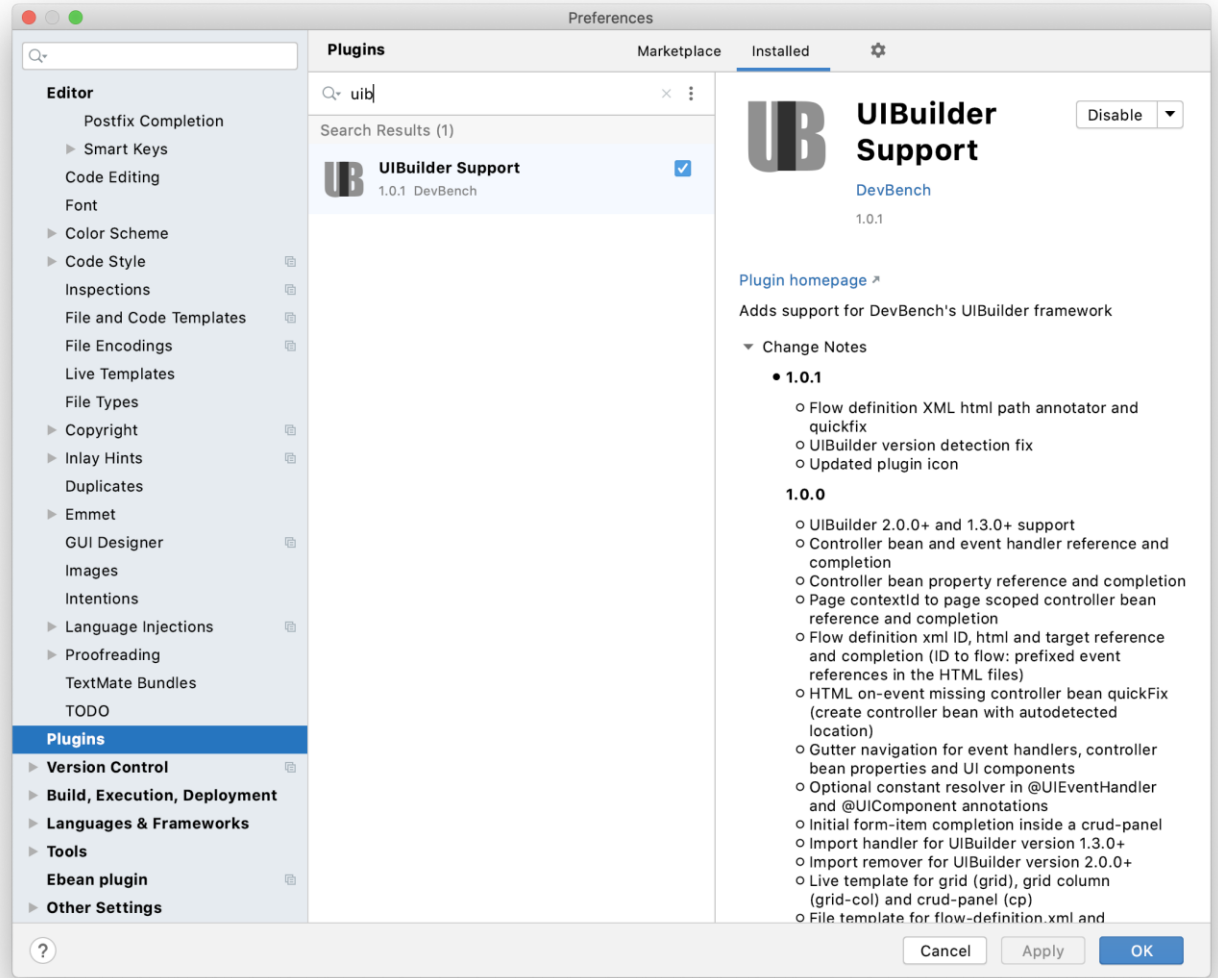
Forráskód: <https://gitlab.com/devbench/uibuilder>

The screenshot shows the GitLab interface for the 'uibuilder' project. The left sidebar contains navigation options: Project overview, Details, Activity, Releases, Repository, Issues (28), Merge Requests (1), Requirements (0), CI / CD, Security & Compliance, Packages, Analytics, Wiki, and Collapse sidebar. The main content area displays the project details for 'uibuilder' (Project ID: 9855404), including 649 Commits, 42 Branches, 9 Tags, 8.1 MB Files, 40.7 GB Storage, and 4 Releases. A merge commit is highlighted: 'Merge branch 'features/deploy\_snapshots\_nightly\_fix' into 'master'' by Majki, authored 3 weeks ago. Below this, there are buttons for README, Apache License 2.0, and CI/CD configuration. A table lists the project's components and their last commit details.

Name	Last commit	Last update
.gitlab/issue_templates	Initial commit	1 year ago
archetypes	Next dev iter 2.0.2-SNAPSHOT	1 month ago
bom	Next dev iter 2.0.2-SNAPSHOT	1 month ago
checkstyle	Resolve "ItemBinding in UIBuilderComboBox seems not to work"	7 months ago
components	Next dev iter 2.0.2-SNAPSHOT	1 month ago



# UIBuilder Jetbrains Idea plug-in



# AppDesigner

- A kollaborációt segíti majd elő a két fejlesztői csapat között
- Integrálja a legacy rendszer fejlesztőit a modernizációs projektbe
- Elősegíti a felületek fejlesztését (WYSIWYG szerkesztő)
- Megkönnyíti a fejlesztés dokumentálását
- Lehetővé teszi, hogy a fejlesztés közös web platformon történjen
- Cloud / private cloud
  - Rolling update
  - Mindig a legfrissebb verzió van mindenhol



# AppDesigner

## Entitás szerkesztő

The screenshot displays the AppDesigner Entity Editor for the 'playground-visualizer' domain. The interface includes a left sidebar with navigation options like 'Processes', 'Actors', 'Glossary', and 'Edit entities'. The main workspace shows a UML class diagram with the following classes and relationships:

- PlaygroundRun** (Entity):
  - Attributes: id: String, data: String, explorations: List<Exploration>
- Exploration** (Entity):
  - Attributes: id: String, playground: PlaygroundRun, playgroundName: String, startTime: LocalDateTime, endTime: LocalDateTime, agentMetrics: List<AgentMetric>, checkpoints: List<Checkpoint>, logs: List<ExplorationLogBase>
- AgentMetric** (Entity):
  - Attributes: id: String, epoch: Integer, frame: Integer, metricName: String, metricValue: Double
- Checkpoint** (Entity):
  - Attributes: id: String, networkName: String, filePath: String
- ExplorationLogBase** (Entity):
  - Attributes: id: String, epoch: Integer, frame: Integer, message: String
- ExplorationError** (Entity):
  - Inherits from ExplorationLogBase.
- ExplorationWarning** (Entity):
  - Inherits from ExplorationLogBase.

Relationships: 'PlaygroundRun' has a 'ManyToOne' relationship with 'Exploration'. 'Exploration' has 'List' relationships with 'AgentMetric', 'Checkpoint', and 'ExplorationLogBase'. 'ExplorationLogBase' has 'Inheritance' relationships with 'ExplorationError' and 'ExplorationWarning'.

The right sidebar shows property configuration for 'playgroundName' (Type: String, Input type: Text field) and 'playground\_name' (Column annotation).

At the bottom, a status bar shows 'No task running' and a Git toolbar with buttons for push, pull, restore, merge, commit, and new branch.

# AppDesigner

## Alkalmazás vezérlés szerkesztő

The screenshot displays the AppDesigner playground-visualizer interface. The domain is set to 'playground-visualizer'. The main workspace shows a UML diagram for a 'Basic Process'. The diagram includes an actor named 'Actor' (labeled '<<Mock Actor>>') connected to a 'playgrounds' component. This component contains a 'viewChart' component and a 'page' component. The 'playgrounds' component is connected to an 'exploration\_chart' component, which has 'back' and 'init' methods. The 'exploration\_chart' component is connected to an 'explorationChartBean' component, which has an '@EventHandler("init")' method. The 'viewChart' component is connected to a 'playgroundsController' component, which has an '@EventHandler("openChart")' method. The interface also features a left sidebar with navigation options like 'Processes', 'Actors', 'Glossary', 'Edit entities', 'Oracle', 'Builder', 'Bundle', 'Flow', 'File editor', 'Theme designer', and 'Project settings'. A 'Nodes' panel on the left lists various components like HTML, Event, MDC, CRUD Panel, Controller Bean, Method, and Window. A 'Diagnostics' panel on the right shows 'Related logs' and 'Filters'. The bottom status bar indicates 'No task running' and provides navigation controls for 'push', 'pull', 'restore', 'merge', 'commit', 'new branch', and 'master'.

# AppDesigner

## WYSIWYG felhasználói felület szerkesztő

Process: "Státusz állítás" File: "statusPage.html"

Utalvány típus megnevezése

Utalvány sorszám

Státusz megnevezése

Névérték

Cnyitás Dátuma

Sorszám lekérdezése

Előző

Következő

Státusz megváltoztatása

Tipus

Sorszám

Névérték

Státusz

Módosító user

Módosítás dátuma

Állomány típus

Intézmény adószám

Components

Search component...

Basic

Layers

Body 6

Vaadin-horizontal-layout 3

Uibuilder-form 1

Grid-builder 9

Uibuilder-window 1

Uibuilder-window 1

Component properties

Settings Style

Classes

- State -

c1511

Selected c1511

General

Float

none left right

Display

Flex

Position

static relative absolute fixed

Top Right

0 px 0 px

Left Bottom

No task running

push pull restore merge commit new branch

master

# AppDesigner

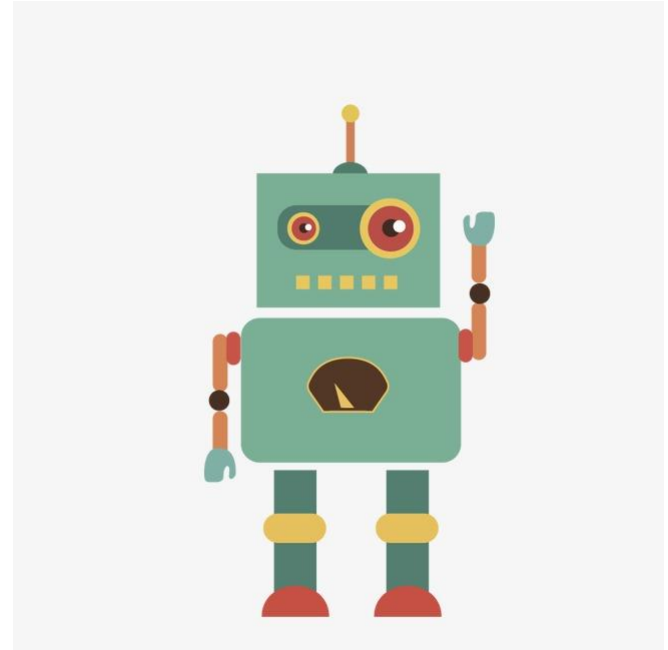
## Beépített eclipse alapú IDE

The screenshot displays the AppDesigner IDE interface. On the left, a sidebar contains navigation options: Processes, Actors, Glossary, Edit entities, Oracle, Builder, Bundle, Flow, File editor (highlighted), Theme designer, and Project settings. The main workspace is divided into three panes. The top-left pane shows a project tree for 'playground-visualizer' with a file named 'PlaygroundsController.java' selected. The top-right pane shows the Java source code for this class, including package declarations, imports, and the implementation of the 'openChart' method. A tooltip is visible over the 'equals' method call in the code, providing details about its behavior: it indicates that the method implements an equivalence relation on non-null object references, and it is both reflexive and symmetric. The bottom status bar shows the progress of the Java Language Server (0% starting) and a Git toolbar with buttons for push, pull, restore, merge, commit, and new branch.

```
1 package eu.redbean.playground.playground_visualizer.controllerbeans.basic_process;
2
3 import com.vaadin.flow.component.notification.Notification;
4 import com.vaadin.flow.component.notification.Notification.Position;
5
6 import eu.redbean.playground.playground_visualizer.model.entities.Exploration;
7 import io.devbench.uibuilder.annotations.ControllerBean;
8 import io.devbench.uibuilder.api.controllerbean.UIComponent;
9 import io.devbench.uibuilder.equals(Object obj) : boolean
10 import io.devbench.uibuilder.getClass() : Class<?>
11 import io.devbench.uibuilder.hashCode() : int
12 import io.devbench.uibuilder.navigate() : void
13 @ControllerBean(value = "explorations", references = {
14     @UIEventHandler(value = "explorations", references = {
15         @wait(), @wait(long timeout) : void
16     })
17     public void openChart(@wait(long timeout) : void
18         if (explorations != null) {
19             Notification.show("Please select an exploration first", 1000, Position.MIDDLE);
20         }
21     }
22 }
23 }
24 }
```

# Forms Plug-in

- Elősegíti a Forms elemek konvertálását az AppDesignerbe egy definiálható mértékben
- Ez nem "silver bullet,, de jelentős segítséget nyújthat a modernizáció felgyorsításában



# Forms Plug-in

DEPTEMP\_fmb.xml [Upload file](#) [Generate entity](#)

[Data Blocks \(2\)](#) [Triggers \(1\)](#) [Program Units \(3\)](#) [LOVs \(2\)](#) [Attached Libraries \(0\)](#) [Form Module Properties](#) [Xml](#)

Search...

[Items \(7\)](#) [DataSource Columns \(4\)](#) [Triggers \(2\)](#) [Relations \(1\)](#) [Data Block Properties](#)

Block Name - DataSource Name	Name	Label Or Prompt	Triggers	DataSource Column	Item Type
DEPARTMENTS - DEPARTMENTS	DEPARTMENT_ID	Department Id	0	<input checked="" type="checkbox"/>	Text Item
EMPLOYEES - EMPLOYEES	DEPARTMENT_NAME	Department Name	0	<input checked="" type="checkbox"/>	Text Item
	LOCATION_ID	Location Id	0	<input checked="" type="checkbox"/>	Text Item
	LOC_STREET		0	<input type="checkbox"/>	Text Item
	MANAGER_ID	Manager Id	0	<input checked="" type="checkbox"/>	Text Item
	MAN_FIRST_NAME		0	<input type="checkbox"/>	Text Item
	MAN_LAST_NAME		0	<input type="checkbox"/>	Text Item

[Item Properties](#) [Triggers](#)

Property Name	Property Value
Distance Between Records	42
Height	14
Item Type	Text Item
Items Display	0
Maximum Length	5
Name	DEPARTMENT_ID
Position X	86.0
Position Y	29.0
Prompt	Department Id

← Collapse Sidebar

push pull restore merge commit new branch master 10



# Forms Plug-in

DEPTEMP\_fmb.xml Upload file Generate entity

Data Blocks (2) Triggers (1) Program Units (3) **LOVs (2)** Attached Libraries (0) Form Module Properties Xml

Name	Title	Name	Return Item	Record Group Name
LOV12		FIRST_NAME	DEPARTMENTS.MANAGER_ID	LOV12
LOV19	Location lov	LAST_NAME	DEPARTMENTS.MAN_FIRST_NAME	1 SELECT ALL EMPLOYEES.EMPLOYEE_ID, EMPLOYEES.FIRST_NAME, EMPLOYEES.LAST_NAME
		EMPLOYEE_ID	DEPARTMENTS.MAN_LAST_NAME	2 FROM EMPLOYEES

Property Name	Property Value
Width	270
Record Group Name	LOV12
Position Y	0.0
Position X	0.0
Name	LOV12
Height	135
Dirty Info	true
Auto Position	true

← Collapse Sidebar

push pull restore merge commit + new branch master

# Köszönöm a figyelmet!

## **További információ, ingyenes konzultáció:**

Gere István:

- [istvan.gere@webvalto.hu](mailto:istvan.gere@webvalto.hu)
- M: +36-30-305-8082