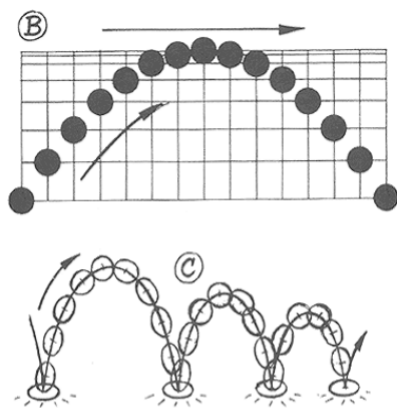
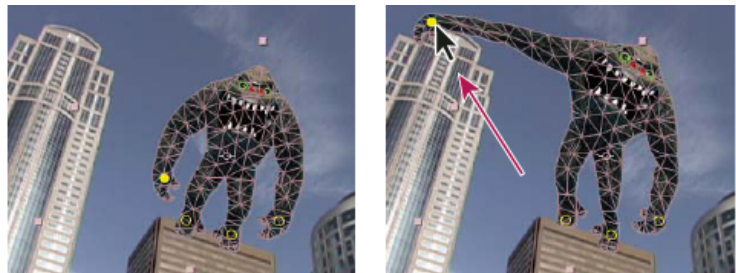


## СПОСОБИ АНІМАЦІЇ ПЕРСОНАЖІВ



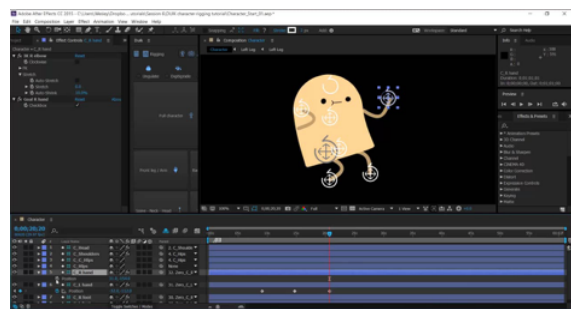
Покадрова анімація



Puppet tool



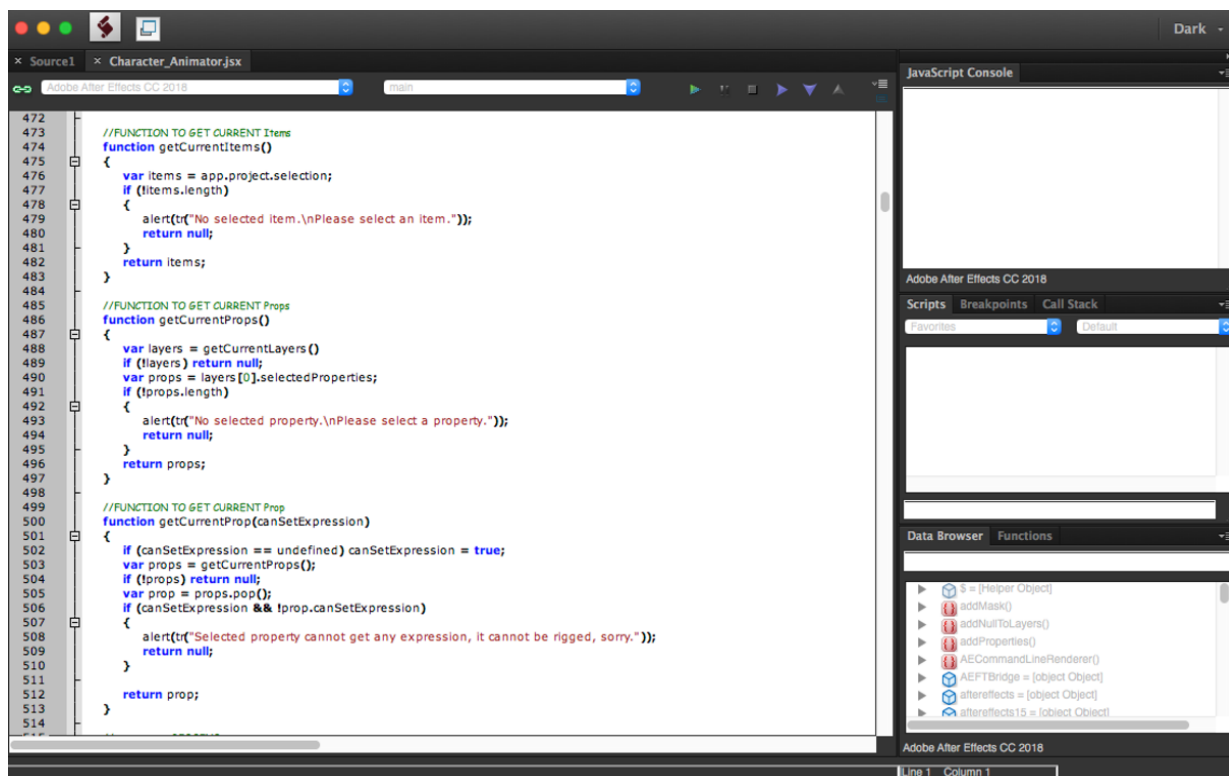
Зв'язки шарів



DUIK

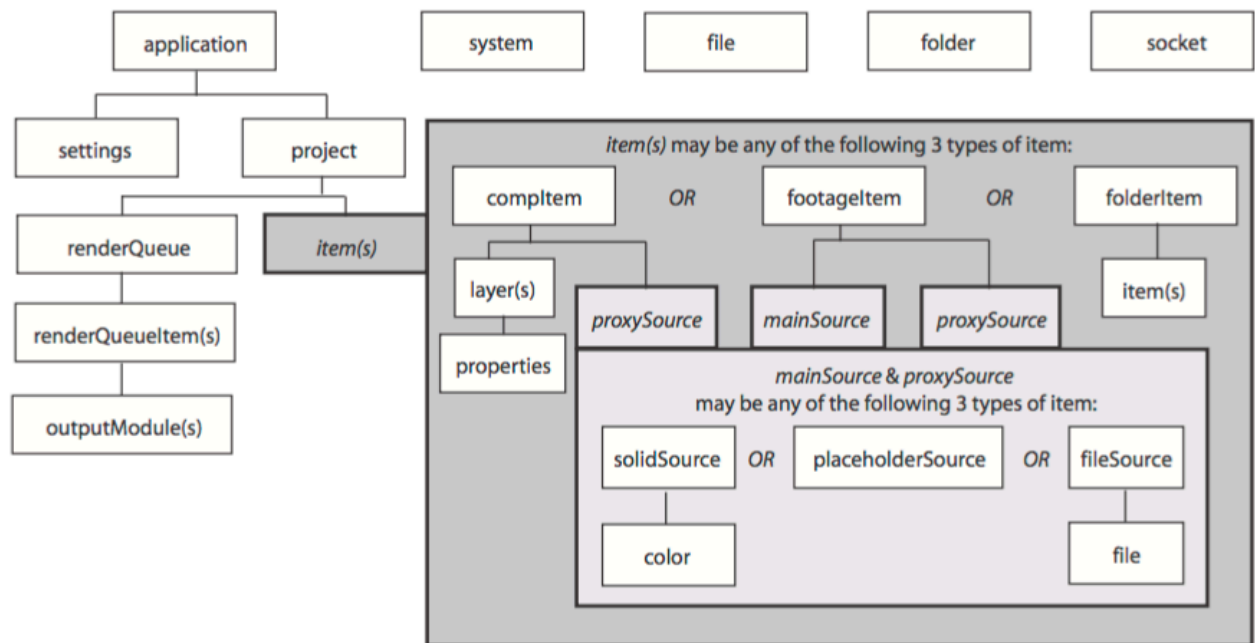
Розроб.	Яровой І.Д.			Способи анімації персонажів	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 1
Затверд.	Гребеннік І.В.			СТ	Листів 9

# ІНТЕРФЕЙС EXTENDSCRIPT TOOLKIT



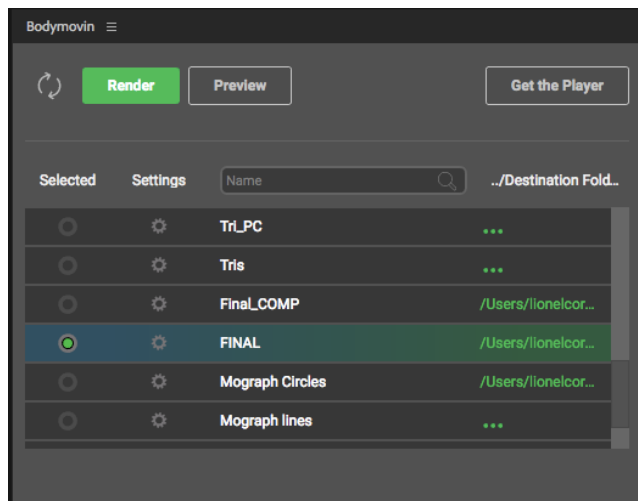
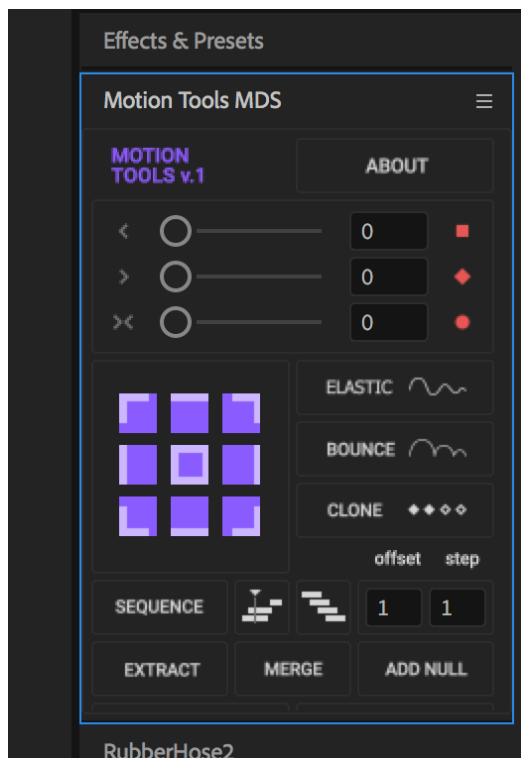
Розроб.	Яровой І.Д.			Інтерфейс ExtendScript Toolkit	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 2
Затверд.	Гребеннік І.В.			СТ	Листів 9

## ІЄРАРХІЯ ОБ'ЄКТІВ В ADOBE AFTER EFFECTS



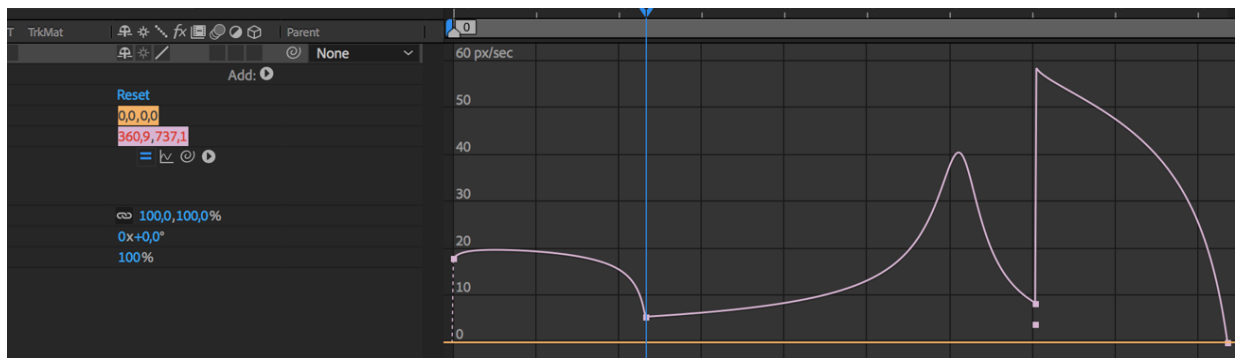
Розроб.	Яровой І.Д.			<i>Ієрархія об'єктів в Adobe After Effects</i>	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 3
Затверд.	Гребеннік І.В.			СТ	Листів 9

## ПРИКЛАД ГРАФІЧНОГО ІНТЕРФЕЙСУ СКРИПТУ В АЕ



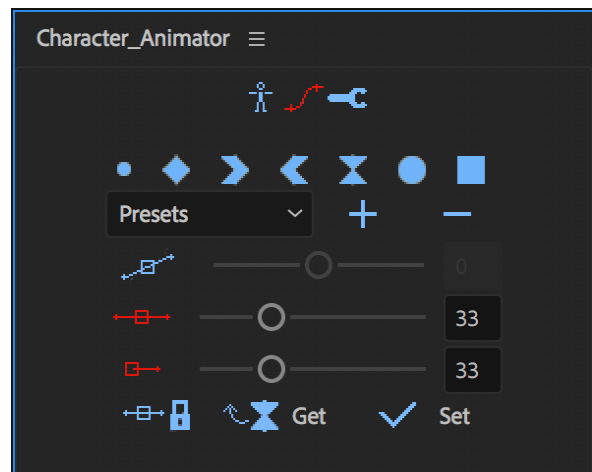
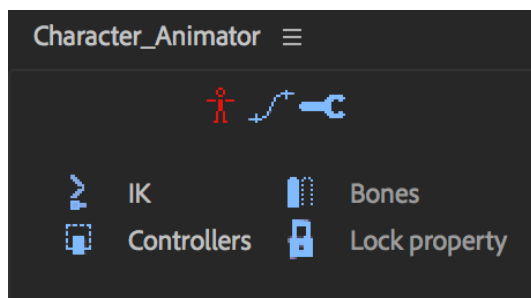
Розроб.	Яровой І.Д.			Приклад графічного інтерфейсу скрипту в АЕ	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 4
Затверд.	Гребеннік І.В.			СТ	Листів 9

## ПРИКЛАД ГРАФІКА ШВИДКОСТІ В ADOBE AFTER EFFECTS



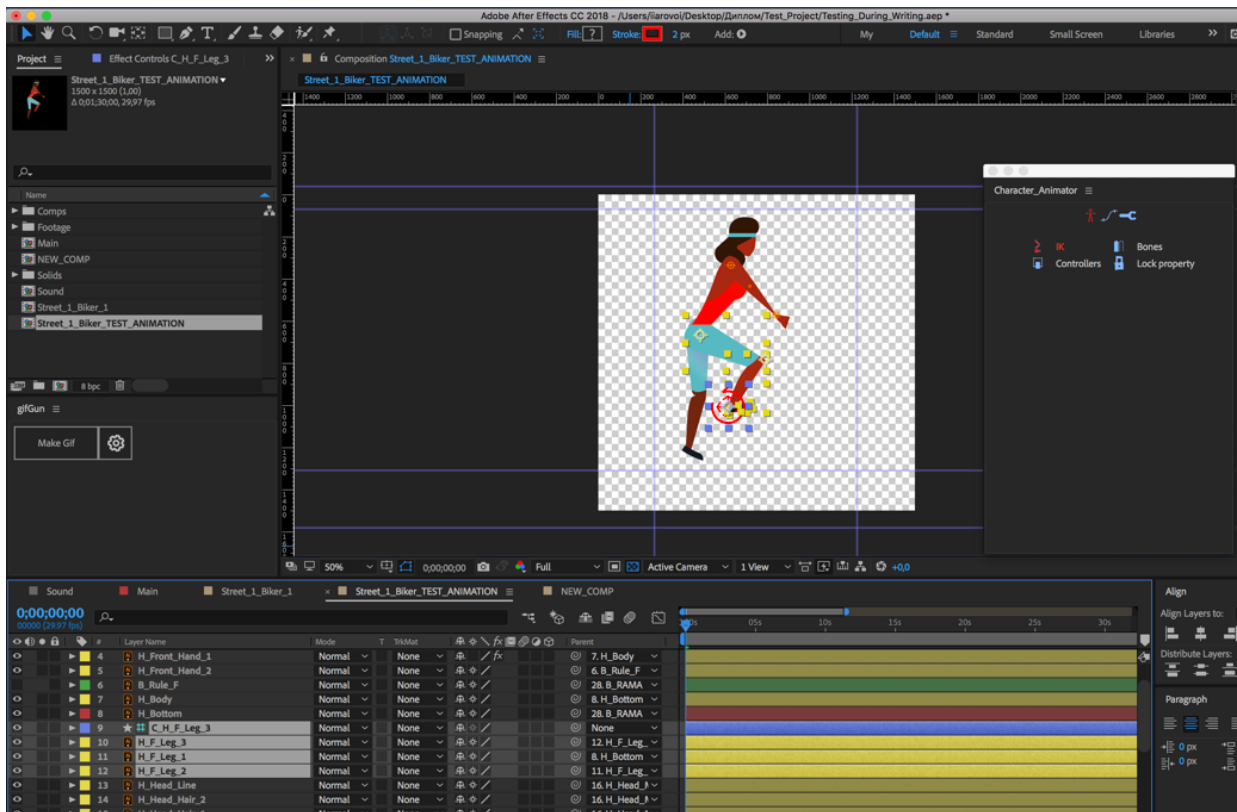
Розроб.	Яровой І.Д.			<i>Приклад графіка швидкості в Adobe After Effects</i>	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 5
Затверд.	Гребеннік І.В.			СТ	Листів 9

# ГРАФІЧНИЙ ІНТЕРФЕЙС РОЗРОБЛЕНОГО СКРИПТУ «CHARACTER ANIMATOR»



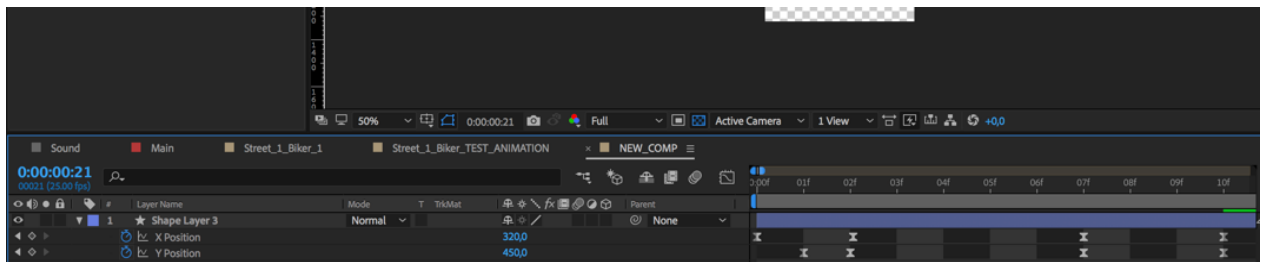
Розроб.	Яровой І.Д.			Графічний інтерфейс розробленого скрипту «Character Animator»	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 6
Затверд.	Гребеннік І.В.			СТ	Листів 9

## ПРОЦЕС СТВОРЕННЯ ІНВЕРСНОЇ КІНЕМАТИКИ



Розроб.	Яровой І.Д.			Процес створення інверсної кінематики	
Перевір.	Калита Н.І.				
Н. Контр.	Калита Н.І.				
				СПР-18-2	Лист 7
Затверд.	Гребеннік І.В.			СТ	Листів 9

## ПРИКЛАД КЛЮЧОВИХ КАДРІВ



<i>Розроб.</i>	<i>Яровой І.Д.</i>			<i>Приклад ключових кадрів</i>	
<i>Перевір.</i>	<i>Калита Н.І.</i>				
<i>Н. Контр.</i>	<i>Калита Н.І.</i>				
				<i>СПР-18-2</i>	<i>Лист 8</i>
<i>Затверд.</i>	<i>Гребеннік І.В.</i>			<i>СТ</i>	<i>Листів 9</i>



## ПРИКЛАД ПЕРСОНАЖІВ АНІМОВАНИХ ЗА ДОПОМОГОЮ РІГУ



<i>Розроб.</i>	<i>Яровой І.Д.</i>			<i>Приклад персонажів анімованих за допомогою рігу</i>	
<i>Перевір.</i>	<i>Калита Н.І.</i>				
<i>Н. Контр.</i>	<i>Калита Н.І.</i>				
				<i>СПР-18-2</i>	<i>Лист 9</i>
<i>Затверд.</i>	<i>Гребеннік І.В.</i>			<i>СТ</i>	<i>Листів 9</i>

ДОДАТОК Б

Текст програми

ГЮИК. 501600.012 – 01 12 01

(позначення документу)

ЗАТВЕРДЖЕНО

ГЮИК. 501600.012 – 01 12 01 – ЛУ

Дослідження методів управління анімацією та їх вдосконалення

Текст програми

ГЮИК. 501600.012 – 01 12 01 – ЛУ

АРКУШІВ 12

2020 р.

Міністерство освіти і науки України  
Харківський Національний Університет Радіоелектроніки

«ЗАТВЕРДЖУЮ»

керівник атестаційної роботи

Калита Н.І.

Дослідження методів управління анімацією та їх вдосконалення

Текст програми

ЛИСТ ЗАТВЕРДЖЕННЯ

ГЮИК. 501600.012 – 01 12 01 – ЛУ

РОЗРОБИВ:

ст. гр. СПР-18-2

Яровой І.Д.

2020р.

```

{
//===== UTILS =====
//FONCTION CARRE
function carre(nombre)
{
    return nombre*nombre;
}
//FUNCTION TO GET CURRENT COMP OR ALERT IF NO CURRENT CompItem
function getCurrentComp()
{
    var comp = app.project.activeItem;
    if (comp instanceof CompItem) return comp;
    else
    {
        alert(tr("Active composition not found.\nPlease select a composition or
activate the composition panel."));
        return null;
    }
}
//FUNCTION TO GET CURRENT Layers
function getCurrentLayers()
{
    var comp = getCurrentComp();
    if (!comp) return null;
    var layers = comp.selectedLayers;
    if (!layers.length
    {
        alert(tr("No selected layer.\nPlease select a layer/property."));
        return null;
    }
    return layers;
}
//FUNCTION TO GET ALL Layers
function getLayers()
{
    var comp = getCurrentComp();
    if (!comp) return null;
    var layers = comp.layers;
    layers = Duik.utils.convertCollectionToArray(layers);
    if (!layers.length)
    {
        alert(tr("No layer in this comp.\nPlease select a composition with
layers."));
        return null;
    }
    return layers;
}
//FUNCTION TO GET CURRENT Items

```

```

function getCurrentItems()
{
    var items = app.project.selection;
    if (!items.length)
    {
        alert(tr("No selected item.\nPlease select an item."));
        return null;
    }
    return items;
}
//FUNCTION TO GET CURRENT Props
function getCurrentProps()
{
    var layers = getCurrentLayers()
    if (!layers) return null;
    var props = layers[0].selectedProperties;
    if (!props.length)
    {
        alert(tr("No selected property.\nPlease select a property."));
        return null;
    }
    return props;
}
//FUNCTION TO GET CURRENT Prop
function getCurrentProp(canSetExpression)
{
    if (canSetExpression == undefined) canSetExpression = true;
    var props = getCurrentProps();
    if (!props) return null;
    var prop = props.pop();
    if (canSetExpression && !prop.canSetExpression)
    {
        alert(tr("Selected property cannot get any expression, it cannot be rigged,
sorry."));
        return null;
    }
    return prop;
}
//===== RIGGING =====
/**
 * Function executed when the IK button is clicked
 *
 * Display the IK options depending on selection and map the click to
 * the correct ik creation function
 */
function ik()
{
    var calques = getCurrentLayers();

```

```
if (!calques) return null;

if (calques.length < 2 || calques.length > 5){
    alert(tr("Select the bones and the controller before creating IK"));
    return;
}
panoik.hide();
ikPanel.show();
//prepIK
app.beginUndoGroup(tr("Duik - prepare IK"));
var ikRig = Duik.utils.prepIK(calques);
app.endUndoGroup();
if (ikRig.type == 0) return;

if (ikRig.type == 1 && ikRig.goal == null){
    ikType1Group.show();
    ikType2Group.hide();
    ikType3Group.hide();
    ikType4Group.hide();
    ik3DGroup.hide();
} else if (ikRig.type == 1 && ikRig.goal != null){
    ikType1Group.hide();
    ikType2Group.show();
    ikType3Group.hide();
    ikType4Group.hide();
    ik1GoalButton.value = true;
    ik2LayerButton.value = false;
    ik3DGroup.enabled = false;
    if (ikRig.threeD){
        ik3DGroup.show();
        ikFrontFacingButton.value = ikRig.frontFacing;
        ikRightFacingButton.value = !ikRig.frontFacing;
    } else {
        ik3DGroup.hide();
    }
} else if (ikRig.type == 2 && ikRig.goal == null){
    ikType1Group.hide();
    ikType2Group.show();
    ikType3Group.hide();
    ikType4Group.hide();
    ik1GoalButton.value = false;
    ik2LayerButton.value = true;
    ik3DGroup.enabled = true;
    if (ikRig.threeD){
        ik3DGroup.show();
        ikFrontFacingButton.value = ikRig.frontFacing;
        ikRightFacingButton.value = !ikRig.frontFacing;
    } else {
```

```
        ik3DGroup.hide();
    }
} else if (ikRig.type == 2 && ikRig.goal != null){
    ikType1Group.hide();
    ikType2Group.hide();
    ikType3Group.show();
    ikType4Group.hide();
    ik3LayerButton.value = false;
    ik2GoalButton.value = true;
    ik3DGroup.enabled = true;
    if (ikRig.threeD){
        ik3DGroup.show();
        ikFrontFacingButton.value = ikRig.frontFacing;
        ikRightFacingButton.value = !ikRig.frontFacing;
    } else {
        ik3DGroup.hide();
    }
} else if (ikRig.type == 3 && ikRig.goal == null){
    ikType1Group.hide();
    ikType2Group.hide();
    ikType3Group.show();
    ikType4Group.hide();
    ik3LayerButton.value = true;
    ik2GoalButton.value = false;
    ik3DGroup.enabled = false;
    if (ikRig.threeD){
        ik3DGroup.show();
        ikFrontFacingButton.value = ikRig.frontFacing;
        ikRightFacingButton.value = !ikRig.frontFacing;
    } else {
        ik3DGroup.hide();
    }
} else if (ikRig.type == 3 && ikRig.goal != null){
    ikType1Group.hide();
    ikType2Group.hide();
    ikType3Group.hide();
    ikType4Group.show();
    ik3DGroup.hide();
}
ikCreateButton.onClick = function() {
    ikRig.frontFacing = ikFrontFacingButton.value;
    if (ikType2Group.visible){
        if (ik2LayerButton.value && ikRig.type == 1){
            ikRig.layer2 = ikRig.goal;
            ikRig.goal = null;
            ikRig.type = 2;
        } else if (ik1GoalButton.value && ikRig.type == 2){
            ikRig.goal = ikRig.layer2;
        }
    }
}
```



```

        ikRig.layer2 = null;
        ikRig.type = 1;
    }
} else if (ikType3Group.visible){
    if (ik3LayerButton.value && ikRig.type == 2){
        ikRig.layer3 = ikRig.goal;
        ikRig.goal = null;
        ikRig.type = 3;
    } else if (ik2GoalButton.value && ikRig.type == 3){
        ikRig.goal = ikRig.layer3;
        ikRig.layer3 = null;
        ikRig.type = 2;
    }
}
app.beginUndoGroup(tr("Duik - IK"));
ikRig.create(ikLockAndShyButton.value);
app.endUndoGroup();
ikPanel.hide();
panoik.show();
}
}
//BEZIER IK
function bikCreateButtonClicked()
{
    var comp = getCurrentComp();
    if (!comp) return null;

    if (comp.selectedLayers.length < 3)
    {
        alert(tr("Select the bones and the controller before creating IK"));
        return;
    }
    app.beginUndoGroup(tr("Duik - Bezier IK"));
    var numCtrls = 1;
    if (bikCubicButton.value) numCtrls = 2;
    Duik.bezierIK(comp.selectedLayers,numCtrls);
    app.endUndoGroup();
    bezierIkPanel.hide();
    panoik.show();
}
//FONCTION QUAND ON CLIQUE SUR GOAL
function pregoal()
{
    var layers = getCurrentLayers()
    if (!layers) return null;
    if (layers.length == 1) {
        //groupe d'annulation
        app.beginUndoGroup(tr("Duik - IK Goal"));
    }
}

```

```

    Duik.goal(layers[0],undefined);
    //groupe d'annulation
    app.endUndoGroup();
    }
    else if (layers.length == 2) {
    //groupe d'annulation
    app.beginUndoGroup(tr("Duik - IK Goal"));
    Duik.goal(layers[0],layers[1]);
    //groupe d'annulation
    app.endUndoGroup();
    }
    else {alert(tr("Select the layer before applying IK Goal"),tr("Attention"),true);}
}
//FONCTION QUAND ON CLIQUE SUR CREER UN CONTROLEUR
function controllerButtonClicked()
{
    if (Duik.settings.controllerType == Duik.layerTypes.VECTOR)
    {
        ctrlPanel.show();
        panoik.hide();
        panointerpo.hide();
    }
    else
    {
        // debut de groupe d'annulation
        app.beginUndoGroup(tr("Duik - ") + tr("Controllers"));
        Duik.addControllers(app.project.activeItem.selectedLayers);
        app.endUndoGroup();
    }
}
function controleur()
{
    var comp = getCurrentComp();
    if (!comp) return null;
    // debut de groupe d'annulation
    app.beginUndoGroup(tr("Duik - ") + tr("Controllers"));
    var newControllers =
    Duik.addControllers(app.project.activeItem.selectedLayers,ctrlAutoLockButton.value,ctrlRotati
onButton.value,ctrlXPositionButton.value,ctrlYPositionButton.value,ctrlScaleButton.value);
    if (ctrlShapeList.selection == 1)
    {
        for (var i=0;i<newControllers.length;i++)
        {
            newControllers[i].arc = true;
            newControllers[i].update();
        }
    }
    else if (ctrlShapeList.selection == 2)

```

```

        {
        for (var i=0;i<newControllers.length;i++)
        {
        newControllers[i].eye = true;
        newControllers[i].update();
        }
        }
        else if (ctrlShapeList.selection == 3)
        {
        for (var i=0;i<newControllers.length;i++)
        {
        newControllers[i].camera = true;
        newControllers[i].update();
        }
        }
        //fin du groupe d'annulation
        app.endUndoGroup();
    }
function ctrlUnlockButtonClicked()
{
    var controllers = Duik.utils.getControllers(app.project.activeItem.selectedLayers);
    app.beginUndoGroup(tr("Duik - Unlock Controllers"));
    for (var i = 0 ; i < controllers.length ; i++)
    {
        controllers[i].unlock();
    }
    app.endUndoGroup();
}
function ctrlLockButtonClicked()
{
    var controllers = Duik.utils.getControllers(app.project.activeItem.selectedLayers);
    app.beginUndoGroup(tr("Duik - Lock Controllers"));
    for (var i = 0 ; i < controllers.length ; i++)
    {
        controllers[i].lock();
    }
    app.endUndoGroup();
}
function ctrlUpdateButtonClicked()
{
    var comp = getCurrentComp();
    if (!comp) return;
    var controllers = Duik.utils.getControllers(comp.selectedLayers);
    app.beginUndoGroup(tr("Duik - Update Controllers"));
    for (var i = 0 ; i < controllers.length ; i++)
    {
        controllers[i].type = Duik.layerTypes.VECTOR;
        controllers[i].color = Duik.settings.controllerColor;
    }
}

```

```

        controllers[i].xPosition = ctrlXPositionButton.value;
        controllers[i].yPosition = ctrlYPositionButton.value;
        controllers[i].rotation = ctrlRotationButton.value;
        controllers[i].scale = ctrlScaleButton.value;
        controllers[i].arc = ctrlShapeList.selection == 1;
        controllers[i].eye = ctrlShapeList.selection == 2;
        controllers[i].camera = ctrlShapeList.selection == 3;

        if (ctrlSizeAutoList.selection.index == 3) controllers[i].size =
Duik.settings.controllerSize;
        else controllers[i].size = 0;
        controllers[i].update();
        controllers[i].unlock();
        if (ctrlAutoLockButton.value)
        {
            controllers[i].lock();
        }
        app.endUndoGroup();
    }
function ctrlUnhideButtonClicked()
{
    var controllers = Duik.utils.getControllers(app.project.activeItem.selectedLayers);
    app.beginUndoGroup(tr("Duik - Unhide Controllers"));
    for (var i = 0 ; i < controllers.length ; i++)
    {
        controllers[i].layer.enabled = true;
    }
    app.endUndoGroup();
}

function ctrlHideButtonClicked()
{
    var controllers = Duik.utils.getControllers(app.project.activeItem.selectedLayers);
    app.beginUndoGroup(tr("Duik - Unhide Controllers"));
    for (var i = 0 ; i < controllers.length ; i++)
    {
        controllers[i].layer.enabled = false;
    }
    app.endUndoGroup();
}

function ctrlResetButtonClicked()
{
    if (!(app.project.activeItem instanceof CompItem)) return;
    var ctrls = app.project.activeItem.selectedLayers;
    if (ctrls.length == 0) ctrls =
Duik.utils.getControllers(app.project.activeItem.layers);
    else ctrls = Duik.utils.getControllers(ctrls);
}

```

```

    app.beginUndoGroup(tr("Reset Ctrl. Transform."));

    for (var i=0;i<ctrls.length;i++)
    {
        var c = ctrl[s[i].layer;
        if (c.parent == null)
c.transform.position.setValue([c.containingComp.width/2,c.containingComp.height/2,0]);
        else c.transform.position.setValue([0,0,0]);
        c.transform.scale.setValue([100,100,100]);
        c.transform.rotation.setValue(0);
        c.transform.opacity.setValue(100);
    }
    app.endUndoGroup();
}
//FONCTION POUR AJOUTER UN (DES) BONE(S)
function bone()
{
    var comp = getCurrentComp();
    if (!comp) return;
    // debut de groupe d'annulation
    app.beginUndoGroup(tr("Duik - Bone"));
    //le(s) calque(s) selectionne(s)
    var calques = comp.selectedLayers ;
    if (calques.length ==0) { alert(tr("Select a pin to create a bone"),tr("Attention"));
return; }

    try { Duik.addBones(calques); }
    catch(e) { alert(e); }
    //fin du groupe d'annulation
    app.endUndoGroup();
}
//LOCK PROPERTY
function lockButtonClicked()
{
    //Prendre l'effet
    var effet = getCurrentProp(true);
    if (!effet) return;
    // debut de groupe d'annulation
    app.beginUndoGroup(tr("Duik - Lock"));
    Duik.lockProperty(effet);
    app.endUndoGroup();
}
//===== ANIMATION =====
//IMPORT RIG
function irRigRefreshButtonClicked()
{
    //list all comps of the project

```

```

        irRigButton.removeAll();
        for (var i = 1 ; i < app.project.numItems;i++)
        {
            if (app.project.item(i) instanceof CompItem)
            {
                irRigButton.add('item',i + '' + app.project.item(i).name);
            }
        }
    }
function irOKButtonClicked()
{
    if (irRigButton.selection == null) return;
    var comp = getCurrentComp();
    if (!comp) return;
    if (irNameText.text == "") alert(tr("You must specify a (unique) name for this
instance of the rig"));
    irPanel.hide();
    panointerpo.show();
    //gets the rig comp
    var index =
parseInt(irRigButton.selection.text.substring(0,irRigButton.selection.text.indexOf(' ')));
    app.beginUndoGroup(tr("Import rig: ") + irNameText.text);
    Duik.importRigInComp(app.project.activeItem,app.project.item(index),irNameText.text)
;
        app.endUndoGroup();
    }
function irDuplicateButtonClicked()
{
    if (irRigButton.selection == null) return;
    if (irNameText.text == "") alert(tr("You must specify a (unique) name for this
instance of the rig"));
    irPanel.hide();
    panointerpo.show();
    //gets the rig comp
    var index =
parseInt(irRigButton.selection.text.substring(0,irRigButton.selection.text.indexOf(' ')));
    app.beginUndoGroup(tr("Duplicate rig: ") + irNameText.text);
    Duik.duplicateRig(app.project.item(index),irNameText.text);
    app.endUndoGroup();

}
//FONCTION WIGGLE OK
function wiggleOKButtonClicked()
{
    // get property
    var prop = getCurrentProp();
    if (!prop) return;
    // debut de groupe d'annulation

```

```

    app.beginUndoGroup(tr("Duik - Wiggle"));
    Duik.wiggle(prop,wiggleSeparate.value);
    //fin du groupe d'annulation
    app.endUndoGroup();
}
//FONCTION WIGGLE
function wiggle()
{
    //regarder le nombre d'axes dans la propriete selectionnee
    var prop = getCurrentProp(true);
    if (prop == undefined) return;
    var dim = prop.propertyValueType ;
    app.beginUndoGroup(tr("Duik - Wiggle"));
    if (dim == PropertyValue.Type.ThreeD_SPATIAL || dim ==
PropertyValueType.ThreeD || dim == PropertyValue.Type.TwoD_SPATIAL || dim ==
PropertyValueType.TwoD)
    {
        panoanimation.hide();
        wigglePanel.show();
    }
    else
    {
        Duik.wiggle(prop);
    }
    app.endUndoGroup();
}
//FONCTION QUI MESURE LE RAYON D'UNE ROUE
function mesurer()
{
    var comp = getCurrentComp();
    if (!comp) return;
    //verifions qu'il y a deux calques selectionnes
    if (app.project.activeItem.selectedLayers.length == 2){
        var dist =
Duik.utils.getDistance(app.project.activeItem.selectedLayers[0],app.project.activeItem.selectedL
ayers[1]);
        rayonbouton.text = dist;
        return dist;
    } else { alert(tr("Select two layers to measure distance from one layer to
another!"),tr("Attention"),true); }

}
//FONCTION QUI CREE UNE ROUE
function roue()
{
    OA = parseFloat(rayonbouton.text);

    if (OA != 0 && OA != NaN)

```

```
        {
            var layers = getCurrentLayers();
            if (!layers) return;
            // debut de groupe d'annulation
            app.beginUndoGroup(tr("Wheel"));
            Duik.wheel(layers[0],OA,roueC.value);
            // fin de groupe d'annulation
            app.endUndoGroup();
        } else { alert (tr("Invalid Radius"),tr("What is the radius of the wheel?"),true); }
    }
//FONCTION LENTILLE
function lentille()
{
    //les calques selectionnes :
    var calques = app.project.activeItem.selectedLayers;
    //verifions qu'il y a plusieurs calques selectionnes
    if (calques.length > 1){
        // debut de groupe d'annulation
        app.beginUndoGroup(tr("Lens Controller"));
        Duik.lensFlare(calques);
        //fin du groupe d'annulation
        app.endUndoGroup();
    } else {
        alert(tr("Select all the layers of the lens flare, beginning by the center"));
    }
}
```



№	Позначення				Найменування	Додаткові відомості	
					Текстові документи		
1.	ГЮИК. 501600.012 ПЗ				Пояснювальна записка	67 стор.	
2.	ГЮИК. 501600.012 – 01 12 01				Текст програми	12 стор.	
					Графічні документи		
5.	ГЮИК. 501600.012 С10				Способи анімації персонажів	1 аркуш	
6.					Інтерфейс ExtendScript Toolkit	1 аркуш	
7.					Ієрархія об'єктів в Adobe After Effects	1 аркуш	
8.					Приклад графічного інтерфейсу скрипту в АЕ	1 аркуш	
9.					Приклад графіка швидкості в Adobe After Effects	1 аркуш	
10.					Графічний інтерфейс розробленого скрипту «Character Animator»	1 аркуш	
11.					Процес створення інверсної кінематики	1 аркуш	
12.					Приклад ключових кадрів	1 аркуш	
13.					Приклад персонажів анімованих за допомогою рігу	1 аркуш	
					ГЮИК. 501600.012 ДЗ		
Змін.	Арк.	№ докум	Підп.	Дата			
<i>Розроб.</i>		<i>Яровой І.Д.</i>			<i>Дослідження методів управління анімацією та їх вдосконалення</i>	Аркуш	Аркушів
<i>Перевір.</i>		<i>Калита Н.І.</i>				1	1
<i>Н. Контр.</i>		<i>Калита Н.І.</i>				<i>ХНУРЕ</i>	
<i>Затв.</i>		<i>Гребеннік І.В.</i>				<i>Кафедра СТ</i>	