

```
package fi.paatti.research.paattiapplication.views.userview;

import com.vaadin.data.Item;
import com.vaadin.data.util.sqlcontainer.SQLContainer;
import com.vaadin.ui.Button.ClickEvent;
import com.vaadin.ui.*;
import fi.paatti.paattidatabasutils.dbsevice.PaattiResearchDBService;
import fi.paatti.paattidatabasutils.names.PaattiColumnNames;
import fi.paatti.paattidatabasutils.names.PaattiTableNames;
import fi.paatti.research.paattiapplication.ElementNames;
import fi.paatti.research.paattiapplication.views.ApplicationTabSheet;
import fi.paatti.research.paattiapplication.views.ApplicationView;
import fi.paatti.research.paattiapplication.views.ApplicationViewTab;

/**
 * The tab shows the current groups of the user.
 */
 * @author Jari Salokangas, jari.p.t.salokangas@student.jyu.fi
 * @author Tapio Keränen, t.tapio.keranen@student.jyu.fi
 */
public class UserGroupInfoTab extends ApplicationViewTab {

    private Table groupTable;
    private Select groupSelect;
    private static final String[] visibleColumns = new String[] {
        PaattiColumnNames.USERGROUP_usergroupID, PaattiColumnNames.USERGROUP_description
    };

    /**
     * Constructor of the class.
     */
    * @param view the parent view.
    */
    public UserGroupInfoTab(ApplicationView view) {
        super(view);
        createLayout();
    }

    /**
     * {@inheritDoc}
     */
    @Override
    protected void createLayout() {
        groupTable = new Table();
    }
}
```

```
groupTable.setSelected(true);
groupTable.setImmediate(true);
groupTable.setSizeFull();
groupTable.setRequired(true);

SQLContainer groupsContainer = view.getDBConnection().getSQLContainerFromDBTable(PaattiTableNames.USERGROUP,
    PaattiColumnNames.USERGROUP_userid, true);

groupSelect = new Select(ElementNames.USER_VIEW_GROUP_SELECT_LABEL, groupsContainer);
groupSelect.setItemCaptionPropertyId(PaattiColumnNames.USERGROUP_name);
groupSelect.setWidth("100%");
groupSelect.setRequired(true);

Button addButton = new Button(ElementNames.USER_VIEW_ADD_GROUP_BUTTON_LABEL);

addButton.addListener(this);
addButton.setWidth("100%");

Button removeButton = new Button(ElementNames.USER_VIEW_REMOVE_GROUP_BUTTON_LABEL);

removeButton.addListener(this);
removeButton.setWidth("100%");

Button openGroupViewButton = new Button(ElementNames.USER_VIEW_OPEN_GROUP_VIEW_BUTTON_LABEL);

openGroupViewButton.addListener(this);
openGroupViewButton.setWidth("100%");

HorizontalLayout groupInfoLayout = new HorizontalLayout();

groupInfoLayout.addComponent(groupSelect);
groupInfoLayout.addComponent(addButton);
groupInfoLayout.setComponentAlignment(groupSelect, Alignment.BOTTOM_LEFT);
groupInfoLayout.setComponentAlignment(addButton, Alignment.BOTTOM_RIGHT);
groupInfoLayout.setMargin(true);
groupInfoLayout.setSpacing(true);
groupInfoLayout.setSizeFull();

VerticalLayout layout = new VerticalLayout();

layout.setMargin(true);
layout.addComponent(groupTable);
layout.addComponent(groupInfoLayout);
layout.addComponent(removeButton);
layout.addComponent(openGroupViewButton);
layout.setComponentAlignment(removeButton, Alignment.BOTTOM_CENTER);
```

```
layout.setComponentAlignment(openGroupViewButton, Alignment.BOTTOM_CENTER);
layout.setExpandRatio(groupTable, 30);
layout.setExpandRatio(groupInfoLayout, 7);
layout.setExpandRatio(removeButton, 1);
layout.setExpandRatio(openGroupViewButton, 1);

setCompositionRoot(layout);
}
/**
 * {@inheritDoc}
 */
@Override
protected void fillTab(Item user) {
    SQLContainer usersGroups = view.getConnection().getUserGroups(user.getItemId());

    groupTable.setContainerDataSource(usersGroups);
    groupTable.setVisibleColumns(visibleColumns);
    setEnabled(true);
}
/**
 * {@inheritDoc}
 */
@Override
protected void clearTab() {
    groupTable.setContainerDataSource(null);
    setEnabled(false);
}
/**
 * {@inheritDoc}
 * <p/>
 * The handled button click events are:
 * <ul>
 * <li>add the user to the selected group</li>
 * <li>remove the user from the selected group, and</li>
 * <li>examine the selected group.</li>
 * </ul>
 */
@Override
public void buttonClick(ClickEvent event) {
    String buttonCaption = event.getButton().getCaption();
    String msg = null;
    PaattiResearchDBService db = view.getConnection();
```

```
if (buttonCaption.equals(ElementNames.USER_VIEW_ADD_GROUP_BUTTON_LABEL)) {
    if (groupSelect.isValid()) {
        msg = db.addUserToGroup(groupSelect.getValue(), view.getSelectedItemId())
            ? ElementNames.UPDATE_OK
            : ElementNames.UPDATE_FAIL;
        view.refresh(view.getSelectedItemId());
    } else {
        msg = ElementNames.SOMETHING_REQUIRED;
    }
} else if (buttonCaption.equals(ElementNames.USER_VIEW_REMOVE_GROUP_BUTTON_LABEL)) {
    if (groupTable.isValid()) {
        msg = db.removeUserFromGroup(groupTable.getValue(), view.getSelectedItemId())
            ? ElementNames.UPDATE_OK
            : ElementNames.UPDATE_FAIL;
        view.refresh(view.getSelectedItemId());
    } else {
        msg = ElementNames.SOMETHING_REQUIRED;
    }
} else if (buttonCaption.equals(ElementNames.USER_VIEW_OPEN_GROUP_VIEW_BUTTON_LABEL)) {
    if (groupTable.isValid()) {
        view.setActiveTab(ApplicationTabSheet.GROUP_TAB, groupTable.getValue());
    } else {
        msg = ElementNames.SOMETHING_REQUIRED;
    }
}
if (msg != null) {
    getWindow().showNotification(msg);
}
}
```