

ДОДАТОК А  
Копія публікації



Міністерство освіти і науки України  
Тернопільський національний економічний університет  
Харківський національний університет радіоелектроніки  
Національний університет «Львівська політехніка»  
Вінницький національний технічний університет  
Асоціація фахівців комп'ютерних інформаційних технологій



МАТЕРІАЛИ

VI Всеукраїнської школи-семінару  
молодих вчених і студентів

# СУЧАСНІ КОМП'ЮТЕРНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Advanced computer information technologies

20-21 травня 2016 р.

THEU  
Тернопіль  
2016



МОБІЛЬНА СИСТЕМА УПРАВЛІННЯ ПРОЦЕСОМ НАВЧАННЯ У ВНЗ Шпінталь М.Я., Гнатівсвич О.В.....	168
МЕТОДИ АСПЕКТНО-ОРІЄНТОВАНОГО ПІДХОДУ В ІНЖЕНЕРІЇ ПРОГРАМНИХ СИСТЕМ Шпінталь М.Я., Журенко А.М.....	169
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ СКЛАДАННЯ РОЗКЛАДУ ІСПИТІВ ЗА ДОПОМОГОЮ ГЕНЕТИЧНИХ АЛГОРИТМІВ Шпінталь М.Я., Резніченко Є.І.....	170
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО РОБОЧОГО МІСЦЯ СУДДІ З КАРТИНГУ Шпінталь М.Я., Резніченко М.І.....	172
МЕТОДИКА ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПРИ ПЕРЕДАЧІ ВЕЛИКИХ ОБ'ЄМІВ ДАНИХ Яркун В.І.....	174

### **КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ**

МОДЕЛЮВАННЯ СТІЙКОЇ СТЕГАНОФОНІЧНОЇ СИСТЕМИ ІЗ ЗАДАНИМИ ХАРАКТЕРИСТИКАМИ МЕРЕЖІ Касянчук М.М., Худько В.Д.....	177
ОСОБЛИВОСТІ ЗАХИСТУ ПЕРСОНАЛЬНИХ МОБІЛЬНИХ ПРИСТРОЇВ ВІД ЗАГРОЗ НЕСАНКЦІОНОВАНОГО ДОСТУПУ ДО ІНФОРМАЦІЇ Сурм'як І.О.....	178
ЗАХИСТ ТА ІДЕНТИФІКАЦІЯ АУДИОМАТЕРІАЛІВ НА ОСНОВІ МЕТОДУ ЦИФРОВОГО ВІДБИТКУ Якименко І.З., Ковтун Н.В.....	179
ЗАХИСТ ДОКУМЕНТІВ НА ОСНОВІ СЕМАНТИЧНОГО АНАЛІЗУ Якименко І.З., Яшук В.Ф.....	181

### **ІНФОРМАЦІЙНО-АНАЛІТИЧНЕ ЗАБЕЗПЕЧЕННЯ ЕКОНОМІЧНОЇ ДІЯЛЬНОСТІ**

ІНТЕРНЕТ-ТЕХНОЛОГІЇ В РЕКЛАМНИХ ПРОЕКТАХ ПІДПРИЄМСТВА Васильків Н.М., Зозуля Р.В., Махно А.Ю., Поліщук Т.О.....	184
МОДЕЛЮВАННЯ ТА БАГАТОВИМІРНИЙ АНАЛІЗ КЛЮЧОВИХ ЕКОНОМІЧНИХ ПОКАЗНИКІВ ІЗ ВИКОРИСТАННЯМ OLAP-ТЕХНОЛОГІЙ Гончар Л.І., Сембрак М.І.....	185
АНАЛІЗ МОДЕЛЕЙ ГЛОБАЛЬНИХ КОНКУРЕНТНИХ ПЕРЕВАГ СУЧАСНИХ ПІДПРИЄМСТВ Добротвор І.Г., Мамлюк Н.О.....	187
МЕТОД ФОРМУВАННЯ ЗАМОВЛЕННЯ НА ОСНОВІ ЧЕРГ З ПРІОРИТЕТОМ Здирко В.В., Струбицька І.П.....	188
АВТОМАТИЗАЦІЯ ФОРМУВАННЯ ТА ОБЛІКУ ЕКСПЕДИТОРСЬКОЇ ДОКУМЕНТАЦІЇ ТРАНСПОРТНОГО ПІДПРИЄМСТВА Курко Ю.В., Магерівський А.В., Вівчар Д.Р.....	189
ВПРОВАДЖЕННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМИ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ В ОРГАНІЗАЦІЇ Лозович Т.М.....	192
МОДЕЛЮВАННЯ РОЗДРІБНОГО РИНКУ КОМП'ЮТЕРНОЇ ТЕХНІКИ Мельник Н.Б.....	193
МЕТОДИ ТА ЗАСОБИ ІДЕНТИФІКАЦІЇ МИТНИХ ПРАВОПОРУШЕНЬ Мороз Б.І., Коноваленко С.М.....	194

УДК 004.91

## МОБІЛЬНА СИСТЕМА УПРАВЛІННЯ ПРОЦЕСОМ НАВЧАННЯ У ВНЗ

Шпінталь М.Я.<sup>1)</sup>, Гнатівич О.В.<sup>2)</sup>

Тернопільський національний економічний університет

<sup>1)</sup>к.т.н., доцент.; <sup>2)</sup>магістрант

### I. Постановка проблеми

Розвиток мобільних пристроїв та нових методів навчання у вищих навчальних закладах (ВНЗ) усе активніше вимагають використання відповідних мобільних додатків. Використання таких мобільних систем управління дозволить підвищити ефективність управлінських методів та методів навчання. Вирішити дану проблему можна шляхом використання електронного журналу успішності. Електронний журнал успішності являє собою інформаційну систему, яка надає можливість краще відслідковувати якість наданої освіти та підвищити її ефективність, що є важливим елементом системи електронного управління освітою[1,2].

### II. Мета роботи

Метою роботи є розробка ефективної автоматизованої мобільної системи управління навчальним процесом у ВНЗ, яка дасть можливість приймати управлінські рішення в режимі реального часу.

Для досягнення мети необхідно вирішити наступні задачі:

- проаналізувати наявні системи з управління навчальним процесом та виявити їх переваги і недоліки;
- розробити методи аналізу навчального процесу в режимі реального часу;
- розроблено методи формування інформаційної моделі для систем автоматизованого управління навчанням;
- розробити та обґрунтувати структуру мобільної автоматизованої системи управління навчальним процесом;
- реалізувати мобільну автоматизовану систему управління навчальним процесом;

### III. Особливості програмної реалізації мобільної системи управління процесом навчання

Взявши за основу розроблені методи аналізу процесу навчання, що базуються на логічних, математичних процедурах в інформаційних технологіях синтезу систем автоматизованого управління навчанням була запропонована наступна технологія реалізації автоматизованої мобільної системи.

Діаграма класів мобільної системи управління процесом навчання у ВНЗ наведена на рисунку 1.



Рисунок 1 - Діаграма класів мобільної системи управління процесом навчання у ВНЗ

1. Клас «Початкове вікно» – даний клас, містить лише 2 основні функції необхідні для входу в систему:

- «увійти як студент» - будуть доступна лише функція перегляду журналу, як загалом так лише за персональними даними, без можливості зміни будь-яких даних журналу;

- «увійти як викладач» - доступні усі функції, що можуть бути надані викладачеві, а саме: перегляд журналу, внесення оцінок, редагування «поля присутності студента», заповнення звіту успішності, вибір дисципліни, вибір навчального семестру, вибір групи, вибір навчального року та «формування персональних навчальних даних».

2. Клас «Викладач» містить поля, що визначають його персональні дані, закріплені за ним дисципліни та пошту. Міститиме також лише 3 функції: «Відкрити журнал» – генерує сторінку на якій виводяться інформація за обраними даними групи студентів, «Зберегти дані профілю», «Вийти» – здійснює вихід викладача із системи.

3. Клас «Студент» аналогічно класу «Викладач» містить поля, що визначають його персональні дані 5 та має наступні функції: «Переглянути журнал», «Вийти» – здійснює вихід студента із системи.

4. Клас Журнал містить необхідні поля для виводу інформації про студента та відповідні функції. Функції, що можуть редагувати чи вносити зміст в БД в якій будуть міститися дані може лише викладач. Функції перегляду, такі як: опис заняття, та вибір дисципліни доступні, як викладачу так і студенту. Функція формування звіту успішності може бути викликана лише викладачем оскільки формує звіт усіх студентів з обраної групи.

Таким чином мобільна інформаційна система «Електронний журнал» дозволить зменшити час роботи викладача над статистичною інформацією навчального процесу, зробити цей процес мобільним, що спростить та підвищить ефективність його роботи.

Розроблено прототип програмного забезпечення, що реалізує дану структуру класів для мобільної платформи Android.

### Висновок

В даній роботі проаналізовано наявні системи з управління навчальним процесом та виявлено їх переваги і недоліки, розроблено методи аналізу навчального процесу в режимі реального часу, розроблено методи формування інформаційної моделі для систем автоматизованого управління навчанням, розроблено та обґрунтовано структуру мобільної автоматизованої системи управління навчальним процесом реалізовано мобільну автоматизовану систему управління навчальним процесом, яка орієнтована на роботу в реальному часі з використанням надійного мережевого зв'язку.

### Список використаних джерел

1. Автоматизована система управління навчальним закладом – MKR. [Електронний ресурс]. - Режим доступу: <http://mkr.org.ua/aboutprojects/index/2>.
2. Антоник Михайло Стефанович. Інформаційна технологія побудови автоматизованої системи управління навчальним процесом: дис. канд. техн. наук: 05.13.06 / Державний комітет зв'язку та інформатизації України ; НАН України; Державний НДІ інформаційної інфраструктури. - Л., 2005. [Електронний ресурс]. - Режим доступу: <http://www.lib.ua-ru.net/diss/cont/21086.html>.

УДК 004.021

## МЕТОДИ АСПЕКТНО-ОРІЄНТОВАНОГО ПІДХОДУ В ІНЖЕНЕРІЇ ПРОГРАМНИХ СИСТЕМ

Шпінталь М.Я.<sup>1)</sup>, Журенко А.М.<sup>2)</sup>

*Тернопільський національний економічний університет*

*<sup>1)</sup>к.т.н., доцент.; <sup>2)</sup> магістрант*

### І. Постановка проблеми

Уже багато років об'єктно-орієнтований підхід до програмування користується широкою популярністю. Саме об'єктно-орієнтовані мови програмування містять все необхідне для того, щоб представити бізнес-логіку проекту в наочному вигляді. Але чи достатньо використання об'єктно-орієнтованого підходу до програмування для того, щоб досягти такої бажаної наочної бізнес-логіки?

## ДОДАТОК Б

## Код програми

```
namespace AttendanceJournalMobileClient.ViewModel
{
    public class MainViewModel : INotifyPropertyChanged
    {
        private bool _isBusy;
        private string _groupshortname { get; set; }
        private bool _layoutenable = true;
        private bool _connection;
        private int _pickerindex = -1;
        private int _teacherjournalpickerindex = -1;
        private bool _teacherjournalmode;
        private int _studentjournalsubjectid;
        private int _teacherjournalgroupid;
        private double _journalStudentEntryLayoutheight = 0;
        private bool _listLayoutvisible;
        private GroupModel _selectedlistgroup;
        private string _selectedgroupinlistsshortname;
        private DateTime _studentjournaldate;
        private DateTime _pickeddate;
        private SubjectModel _subjectvm;
        private GroupModel _groupvm;
        private string _teachergroupshortname;
        private BllSubjectModel _ajrootselectedsubject;
        private bool _isonline;
        private BllJournalModel _journalModel;
        private LoginModel _loginvm;
        private UserModel _uservm;
        private DisplayUserModelforGroupsList _displayuser;
        private BllGroupModel _teacherjournalgroupinfo;
        private int _selectedcourse;
        private DateTime _teacherlessontime;
        private TimeSpan _teacherlesstime;

        private ObservableCollection<DisplayUserModelforGroupsList> _teacherjournalusercollection;
        private ObservableCollection<BllJournalModel> _collectionjrecords;
```

```

private ObservableCollection<BllGroupModel> _collectiongroups;
private ObservableCollection<BllSubjectModel> _collectionteacherssubjects;
private ObservableCollection<BllSubjectModel> _collectionstudentssubjects;
private ObservableCollection<DisplayUserModelforGroupsList> _collectionusers;
private ObservableCollection<DisplayUserModelforGroupsList> _collectionofteachers;
private ObservableCollection<DisplayJournalrecrodsStudent> _studentjournalrecords;
private ObservableCollection<TeacherJournalDisplayModel> _teacherjournaldisplaywritecollection;

```

```

public ObservableCollection<BllJournalModel> testcollection { get; set; }

```

```

public MainViewModel()
{
    _uservm = new UserModel();
    _loginvm = new LoginModel();
    _subjectvm = new SubjectModel();
    _groupvm = new GroupModel();
    _firstvmjournal = new JournalModelFirstCourse();
    _secondvmjournal = new JournalModelSecondCourse();
    _thirdvmjournal = new JournalModelThirdCourse();
    _fourthvmjournal = new JournalModelFourhCourse();
    _fifthvmjournal = new JournalModelFifthCourse();
    _journalModel = new BllJournalModel();
}

```

```

public bool LayoutEnable
{
    get { return _layoutenable; }
    set
    {
        if (_layoutenable != value)
        {
            _layoutenable = value;
            OnPropertyChanged(nameof(LayoutEnable));
        }
    }
}

```

```

public bool IsOnline => DependencyService.Get<IInternet>().CheckInternetConnection();

```

```

//LOGIN BLOCK
public Command LoginCommand
{
    get
    {
        return new Command( () =>
        {
            if (DependencyService.Get<IInternet>().CheckInternetConnection())
            {
                _isBusy = true;
                _layoutenable = false;
                OnPropertyChanged(nameof(IsBusy));
                OnPropertyChanged(nameof(LayoutEnable));
                if (IsLoginSuccess().Result)
                {
                    if (GetGroupData().Result)
                    {
                        _isBusy = false;
                        _layoutenable = true;
                        Xamarin.Forms.Application.Current.MainPage = new NavigationPage(new
AJStudentRootPage());
                    }
                    else
                    {
                        Xamarin.Forms.Application.Current.MainPage = new NavigationPage(new AJRootPage());
                    }
                }
                else
                {
                    Alerter("Some Input Data incorrect");
                }
            }
            else
            {
                Alerter("No internet connection");
            }
        }
    });
}
}

```

```

public bool HasValidInput => _loginvm.Login.Length < 20 && _loginvm.Login.Length > 5 &&
_loginvm.Password.Length < 20 && _loginvm.Password.Length > 5 &&
    !string.IsNullOrEmpty(_loginvm.Role) && _loginvm.Role.Length == 7;

```

```

public async Task<bool> IsLoginSuccess()
{
    using (HttpClient client = new HttpClient())
    {
        string uri = "http://localhost:54290/Users/MibileLoginUser";
        var str = JsonConvert.SerializeObject(_loginvm);
        HttpContent httpcontent = new StringContent(str);
        httpcontent.Headers.ContentType = new MediaTypeWithQualityHeaderValue("application/json");
        var result = await client.PostAsync(uri, httpcontent);

        if (!result.IsSuccessStatusCode) return false;
        using (var data = result.Content)
        {
            var stringdata = await data.ReadAsStringAsync();
            UserModelProp = JsonConvert.DeserializeObject<UserModel>(stringdata);
            return true;
        }
    }
}

```

```

public async Task<bool> GetGroupData()
{
    if (_uservm.Group == 0) return false;
    using (HttpClient client = new HttpClient())
    {
        string uri = "http://localhost:54290/Groups/ID/" + $"{_uservm.Group}";
        var result = await client.GetAsync(uri);

        if (!result.IsSuccessStatusCode) return false;
        using (var data = result.Content)
        {
            var stringdata = await data.ReadAsStringAsync();
            GroupModelProp = JsonConvert.DeserializeObject<GroupModel>(stringdata);
            return true;
        }
    }
}

```



```
    }  
}  
  
public string VmLogin  
{  
    get { return _loginvm.Login; }  
    set  
    {  
        if (_loginvm.Login != value)  
        {  
            _loginvm.Login = value;  
            OnPropertyChanged(nameof(VmLogin));  
            OnPropertyChanged(nameof(IsValidInput));  
        }  
    }  
}  
  
public string VmLoginPassword  
{  
    get { return _loginvm.Password; }  
    set  
    {  
        if (_loginvm.Password != value)  
        {  
            _loginvm.Password = value;  
            OnPropertyChanged(nameof(VmLoginPassword));  
            OnPropertyChanged(nameof(IsValidInput));  
        }  
    }  
}  
  
public string VmLoginRole  
{  
    get { return _loginvm.Role; }  
    set  
    {  
        if (_loginvm.Role != value)  
        {  
            _loginvm.Role = value;  
            OnPropertyChanged(nameof(VmLoginRole));  
        }  
    }  
}
```

```

        OnPropertyChanged(nameof(IsValidInput));
    }
}

public bool IsBusy
{
    get { return _isBusy; }
    set
    {
        _isBusy = value;
        OnPropertyChanged(nameof(IsBusy));
    }
}
//END LOGIN BLOCK

public UserModel UserModelProp
{
    get { return _uservm; }
    set
    {
        if (_uservm != value)
        {
            _uservm = value;
            OnPropertyChanged(nameof(UserModelProp));
        }
    }
}

public GroupModel GroupModelProp
{
    get { return _groupvm; }
    set
    {
        if (_groupvm != value)
        {
            _groupvm = value;
            OnPropertyChanged(nameof(GroupModelProp));
            UserGroup();
            OnPropertyChanged(nameof(UserGroupShortName));
        }
    }
}

```

```

    }
}

public string UserBirthday => $"{_uservm.UserBirthday.Date.ToString("d")}";

public void UserGroup()
{
    string[] output = GroupModelProp.GroupName.Split(' ');
    foreach (var item in output)
    {
        _groupshortname = item[0].ToString();
    }
}

public string UserGroupShortName
{
    get { return _groupshortname; }
    set
    {
        if (_groupshortname != value)
        {
            _groupshortname = value;
            OnPropertyChanged(nameof(UserGroupShortName));
        }
    }
}

public Command JournalsCommandStudent
{
    get
    {
        return new Command(async () =>
        {
            _isBusy = true;
            _layoutenable = false;
            OnPropertyChanged(nameof(IsBusy));
            OnPropertyChanged(nameof(LayoutEnable));
            if (GetJournalRecordsAtDate().Result)

```

```

    {
        if (GetTeachersList().Result)
        {
            if (GetAllSubjects().Result)
            {
                ConvertJournaltoDiplsayformat();
                _isBusy = false;
                _layoutenable = true;
                await Xamarin.Forms.Application.Current.MainPage.Navigation.PushAsync(new
StudentJournalPage());
            }
        }
    }
});
}
}

```

```

public void ConvertJournaltoDiplsayformat()
{
    var lectorinitials = "";
    var tempjournalsrec = new ObservableCollection<DisplayJournalrecrodsStudent>();
    for (int i = 0; i < _collectionjrecords.Count; i++)
    {
        if (_collectionjrecords[i].LessonSubject == _collectionstudentsubjects[i].SubjectId)
        {
            lectorinitials = GetTeacherInitials(_collectionstudentsubjects[i].SubjectId);
        }
        break;
    }
    for (int i = 0; i < _collectionjrecords.Count; i++)
    {
        tempjournalsrec[i].LessonDate = _collectionjrecords[i].LessonDate;
        tempjournalsrec[i].LessonName = _collectionstudentsubjects[i].SubjectName;
        tempjournalsrec[i].LessonMark = _collectionjrecords[i].LessonMark;
        tempjournalsrec[i].Lessonlector = lectorinitials;
    }

    _studentjournalrecords = tempjournalsrec;
}

```

```

public string GetTeacherInitials(int id)
{
    string lector = "";
    int lectorid = 0;
    for (int i = 0; i < _collectionstudentssubjects.Count; i++)
    {
        if (_collectionstudentssubjects[i].SubjectLector == id)
        {
            lectorid = id;
        }
    }
    for (int i = 0; i < _collectionofteachers.Count; i++)
    {
        if (_collectionofteachers[i].UserId == lectorid)
        {
            lector = $"{_collectionofteachers[i].UserSurname}_{_collectionofteachers[i].UserFirstName}";
        }
    }
    return lector;
}

public int GetTeacherId()
{
    int lectorid = 0;
    for (int i = 0; i < _collectionstudentssubjects.Count; i++)
    {
        if (_collectionjrecords[i].LessonSubject != _collectionstudentssubjects[i].SubjectId) continue;
        lectorid = _collectionstudentssubjects[i].SubjectLector;
        return lectorid;
    }
    return lectorid;
}

public async Task<bool> GetTeachersList()
{
    using (HttpClient client = new HttpClient())
    {
        string uri = "http://localhost:54290/Users/0";

        var result = await client.GetAsync(uri);
    }
}

```

```

        if (result.IsSuccessStatusCode)
        {
            using (HttpContent data = result.Content)
            {
                var stringdata = await data.ReadAsStringAsync();
                _collectionofteachers
                =
                JsonConvert.DeserializeObject<ObservableCollection<DisplayUserModelforGroupsList>>(stringdata);
                return true;
            }
        }
        return false;
    }
}

public async Task<bool> GetAllSubjects()
{
    using (HttpClient client = new HttpClient())
    {
        string uri = "http://localhost:54290/Subjects/Course/"+${_groupvm.GroupCourse}";

        var result = await client.GetAsync(uri);

        if (result.IsSuccessStatusCode)
        {
            using (HttpContent data = result.Content)
            {
                var stringdata = await data.ReadAsStringAsync();
                _collectionstudentsubjects
                =
                JsonConvert.DeserializeObject<ObservableCollection<BllSubjectModel>>(stringdata);
                return true;
            }
        }
        return false;
    }
}

public async Task<bool> GetJournalRecordsAtDate()
{
    using (HttpClient client = new HttpClient())
    {
        string uri = "";
    }
}

```

```

switch (_groupvm.GroupCourse)
{
    case 1:
        uri =
"http://localhost:54290/FirstJournalEntry/"+${StudentSubjectJournalId}/+{DateFromPicker.Date}";
        break;
    case 2:
        uri =
"http://localhost:54290/SecondJournalEntry/"+${StudentSubjectJournalId}/+{DateFromPicker.Date}";
        break;
    case 3:
        uri =
"http://localhost:54290/ThirdJournalEntry/"+${StudentSubjectJournalId}/+{DateFromPicker.Date}";
        break;
    case 4:
        uri =
"http://localhost:54290/FourthJournalEntry/"+${StudentSubjectJournalId}/+{DateFromPicker.Date}";
        break;
    case 5:
        uri =
"http://localhost:54290/FifthJournalEntry/"+${StudentSubjectJournalId}/+{DateFromPicker.Date}";
        break;
    default:
        break;
}

var result = await client.GetAsync(uri);

if (result.IsSuccessStatusCode)
{
    using (HttpContent data = result.Content)
    {
        var stringdata = await data.ReadAsStringAsync();
        _collectionjrecords =
JsonConvert.DeserializeObject<ObservableCollection<BllJournalModel>>(stringdata);
        return true; }
    }
return false;
}
}

```

```
public bool IsValidJournalStudentButton => !DateFromPicker.Equals(null) && StudentSubjectJournalId >= 1;
```

```
public DateTime DateFromPicker
{
    get { return _pickeddate; }
    set
    {
        if(_pickeddate != value)
        {
            _pickeddate = value;
            OnPropertyChanged(nameof(DateFromPicker));
        }
    }
}
```

```
public int SelectIndexPicker
{
    get { return _pickerindex; }
    set
    {
        if(_pickerindex != value)
        {
            _pickerindex = value;
            OnPropertyChanged(nameof(SelectIndexPicker));
        }
    }
}
```

```
public int StudentSubjectJournalId
{
    get { return _studentjournalsubjectid; }
    set
    {
        if(_studentjournalsubjectid != value)
        {
            _studentjournalsubjectid = value;
            OnPropertyChanged(nameof(StudentSubjectJournalId));
        }
    }
}
```



```

public DateTime StudentJournalDate
{
    get { return _studentjournaldate; }
    set
    {
        if (_studentjournaldate != value)
        {
            _studentjournaldate = value;
            OnPropertyChanged(nameof(StudentJournalDate));
        }
    }
}

public double JournalStudentEntryLayoutheight
{
    get { return _journalStudentEntryLayoutheight; }
    set
    {
        if (!_journalStudentEntryLayoutheight.Equals(value))
        {
            _journalStudentEntryLayoutheight = value;
            OnPropertyChanged(nameof(JournalStudentEntryLayoutheight));
        }
    }
}

public Command ChangeJournalStudentEntryLayoutheight
{
    get
    {
        return new Command(() =>
        {
            JournalStudentEntryLayoutheight = JournalStudentEntryLayoutheight.Equals(0) ? 160 : 0;
        });
    }
}

public bool IsValidJournalTeacherButton => _pickerindex > -1 && _pickerindex < 5 &&
_teacherjournalgroupid >= 0 && IsOnline;

```