

Iskolendar

Software Architectural Design

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:

Espino, Christel Anne A.
Sanchez, Patrick Joseph A.
Tanquintic, Allure Migy Daidelle R.

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2018-2019



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

Unique Reference:

The documents are stored in the <https://github.com/christelespino/iskolendar> referenced with *Iskolendar - Architectural Design.pdf*.

Purpose:

The purpose of this document is to show the revised software architecture of the application with finalized packages encapsulating the Boundary Classes, Control Classes, and Entity Classes.

Audience:

- UP Diliman College of Engineering organizations who need publicity for their events and announcements
- UP Diliman students who wish to be informed of the organizations' events

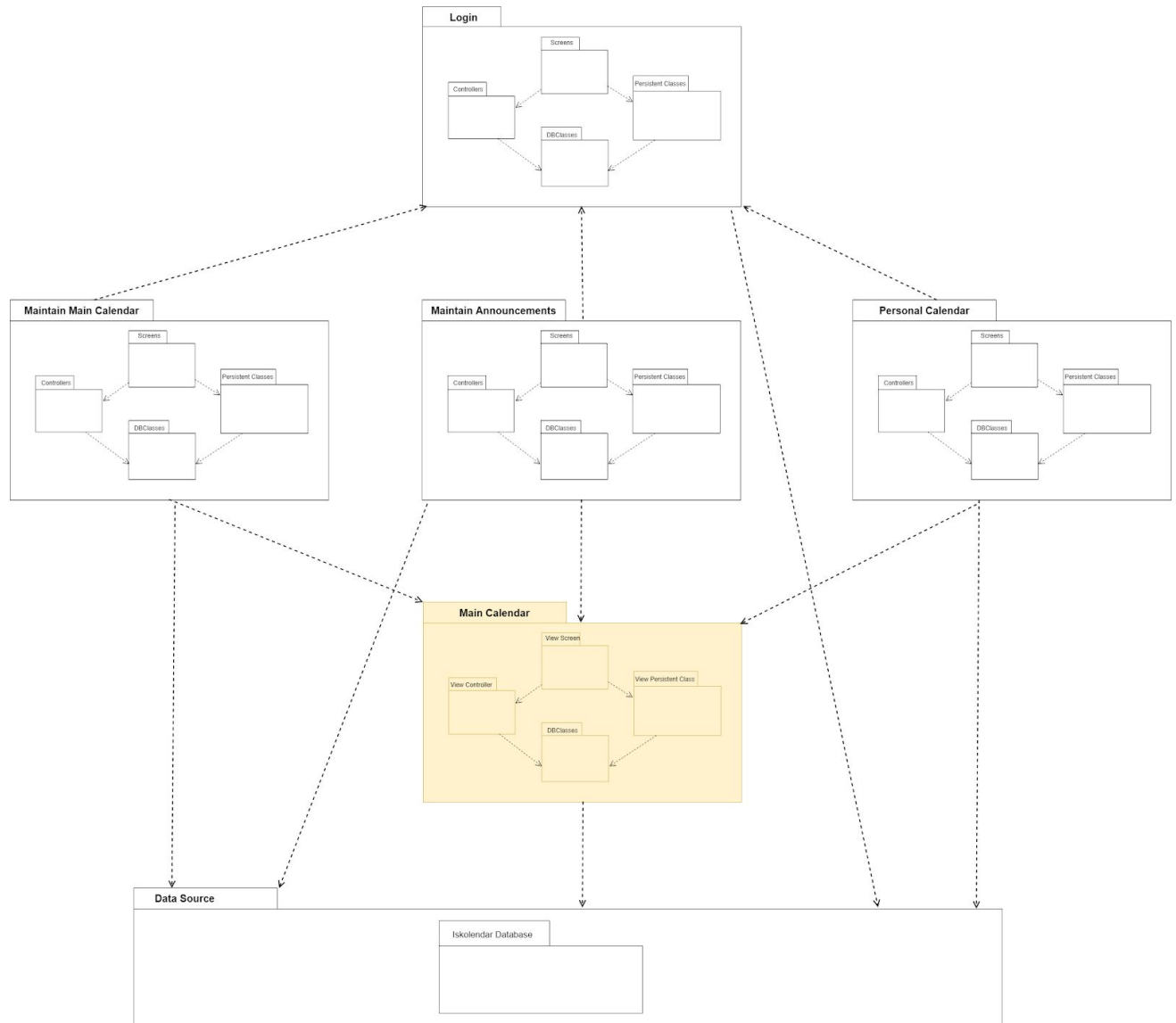
Revision Control:

| <i>Revision Date</i> | <i>Person Responsible</i> | <i>Version Number</i> | <i>Contribution/Modification</i> |
|-----------------------------|----------------------------------|------------------------------|--|
| 11/04/2018 | Christel Espino | 1.0 | Initial Document, Added system name and description, Added software architecture model draft, user interface packages, DAO packages, transfer objects packages, data source packages |
| 11/05/2018 | Patrick Joseph Sanchez | 1.1 | Added Descriptions in Transfer Objects Package. |
| 11/06/2018 | Allure Tanquintic | 1.2 | Added Descriptions in User Interface Packages. |
| 11/06/2018 | Patrick Joseph Sanchez | 2.0 | Added Descriptions in Data Access Objects Packages. Added Controllers Package. |

System Name: Iskolendar

Description: Iskolendar is a simple, centralized, calendar-like application where student organizations from the UP College of Engineering can post and promote their events such as seminars, workshops, review sessions, and the like so that students can view everything in one place. Student users may also view a personal calendar where only the events they're interested in joining will be displayed so that they will be reminded of important dates. On the other hand, the system will be beneficial to the organizations when it comes to event publicity and information dissemination.

Revised Software Architecture Model:



User Interface Package:

| Screen Name | Description |
|-------------------------|--|
| LoginUI | <p>This is the screen where the user logs in to the system.</p> <p><i>Attributes:</i> private int studentID; private String username; private String password; ...</p> <p><i>Responsibilities:</i> public void enterStudentData(int studentID, String username, String password); public void enterOrgLeaderData();</p> |
| MaintainMainCalendarUI | <p>This is the screen that the engg org leaders see and interact with after logging into the system.</p> <p><i>Attributes:</i> private Event[] event; ...</p> <p><i>Responsibilities:</i> public void displayEvents();</p> |
| CreateEventUI | <p>This is the screen that enables the users to enter the details of the event they wish to create.</p> <p><i>Attributes:</i> private Event[] events; ...</p> <p><i>Responsibilities:</i> public void enterEventDetails(int EventID, String title, String description, String dateStart, String dateEnd, String timeStart, String timeEnd, String venue, String type, int lim); public void createEvent(Event event)</p> |
| EditEventUI | <p>This is the screen that enables the users to edit the details of the event they created.</p> <p><i>Attributes:</i> private Event[] events ...</p> <p><i>Responsibilities:</i> public void editDetails(int EventID, String title, String description, String dateStart, String dateEnd, String timeStart, String timeEnd, String venue, String type, int lim)</p> |
| DeleteEventUI | <p>This is the screen that enables the user to delete the event they created.</p> |
| MaintainAnnouncementsUI | <p>This is the screen that the engg org leaders see and interact with the announcements they had created or posted.</p> |

| | |
|-----------------------|---|
| | <i>Attributes:</i> private Announcement[] announcement; ... <i>Responsibilities:</i> public void displayAnnouncements(); |
| CreateAnnouncementUI | This is the screen that enables the users to enter the details of the announcement they wish to create. <i>Attributes:</i> private Event[] events ... <i>Responsibilities:</i> private void enterAnnouncementDetails(int AnnouncementID, String title, String description) |
| EditAnnouncementUI | This is the screen that enables the users to edit the details of the announcement they created. <i>Responsibilities:</i> public void editAnnouncement(Announcement[] announcements, Announcement newAnnouncement); |
| Delete AnnouncementUI | This is the screen that enables the user to delete the announcement they created. |
| PersonalCalendarUI | This is the screen where the student users can see the events they had joined or bookmarked and the announcements that is related to the events. <i>Attributes:</i> private Event[] event ... <i>Responsibilities:</i> public void displayEventsPC(); |
| MainCalendarUI | This is the screen where the student users can see the events posted by the engg org leaders. <i>Attributes:</i> private Event[] event ... <i>Responsibilities:</i> public void displayEventsMC(); |
| ViewEventUI | This is the screen that the student user sees when they want to view an event to see its details. <i>Attributes:</i> private Event[] event ... <i>Responsibilities:</i> public void displayEventDetails(Event[] events); |
| AnnouncementsUI | This is the screen where the student users can see all the announcements. |

| | |
|--------------------|--|
| | <i>Attributes:</i> private Announcement[] announcement; ... Responsibilities: public void displayAnnouncements(Announcement[] announcement); |
| ViewAnnouncementUI | This is the screen where the student users can see all the announcements. <i>Attributes:</i> private Announcement[] announcement; private Announcement a; ... Responsibilities: public void displayAnnDetails(Announcement a); |

Controllers Package:

| Class Name | Description |
|-------------------------------|--|
| ValidateUserController | This is the control that verifies the login details entered by the user. <u>Responsibilities:</u> public void validateUser(String username, String password) |
| DisplayMainCalendarController | This is the control that displays the Main Calendar. <u>Responsibilities:</u> public MainCalendar viewMainCalendar() public Event viewDateEvents() public Event viewGoing() public Event getGoingData() |
| AddEventController | This is the control that adds an Event to the system. <u>Responsibilities:</u> addEvent(Event a, MainCalendar a) |
| EditEventController | This is the control that updates the Event chosen from the system. <u>Responsibilities:</u> editEvent(Event a, MainCalendar a) |
| DeleteEventController | This is the control that deletes an Event from the system. <u>Responsibilities:</u> removeEvent(Event a) |
| AddAnnouncementController | This is the control that posts announcements about the changes in the events. <u>Responsibilities:</u> AddAnnouncement(Event a) |
| JoinEventController | This is the control that adds an Event to the student's personal calendar. <u>Responsibilities:</u> public void AddEvent(Event a, StudentCalendar a, String studentNumber) |
| AddParticipantController | This is the control that adds a Student to the event's list of participants. <u>Responsibilities:</u> public void AddParticipant(Event a, Student a) |

Data Access Objects Packages:

| DAO Name | Description |
|-------------------|---|
| DBStudent | <p>This data access object is responsible for getting student data from a database.</p> <p><u>Attributes:</u></p> <p>private Athlete thestudent; private Guardian theGuardian; private AthleteList theAthleteList;</p> <p><u>Methods:</u></p> <p>private void connectAthleteDatabase(String URL, String user, String password); private void insertAthlete(Athlete theAthlete); private void updateAthlete(Athlete theAthlete); private void deleteAthlete(Athlete theAthlete);</p> |
| DBOrgLeader | This data access object is responsible for getting org leader data from a database. |
| DBEvent | This data access object is responsible for getting data of a certain event from a database. |
| DBAnnouncement | This data access object is responsible for getting data of announcements from a database. |
| DBStudentEvent | This data access object is responsible for getting data of the events a student has joined for the personal calendar, and the participants for each event for the main calendar from a database. |
| DBStudentBookmark | This data access object is responsible for getting data of the events a student has bookmarked for the personal calendar. |

Transfer Objects Package:

| Class Name | Description |
|-------------------|---|
| Student | This is the entity class student, which contains the data about the student. |
| PCLStudent | <p>This is the entity class list of student, which contains all of the student.</p> <p><u>Attributes:</u></p> <p>private Student[] students</p> <p><u>Methods:</u></p> <p>private void addStudents(Student s)</p> <p>private Student searchStudents(int s_id)</p> |
| OrgLeader | <p>This is the entity class org leader, which contains the data about the org leader.</p> <p><u>Attributes:</u></p> <p>private String SN_eol</p> <p>private String username</p> <p>private String password</p> <p>private String lastname</p> <p>private String firstname</p> <p>private String position</p> <p>private String organization</p> <p>private String orgAcronym</p> <p>private String DegreeProgram</p> <p>private int year</p> <p>private String emailAd</p> <p><u>Methods:</u></p> <p>private void set studentID(int theID);</p> <p>private void setLastname(String theName);</p> <p>private void setFirstname(String theName);</p> <p>...</p> <p>private int getstudentID();</p> <p>private String getLastname();</p> <p>private String getFirstname();</p> |
| PCLOrgLeader | <p>This is the entity class list of org leader, which contains all of the org leader.</p> <p><u>Attributes:</u></p> <p>private OrgLeadert[] leaders</p> <p><u>Methods:</u></p> <p>private void addOrgLeader(OrgLeader o)</p> <p>private OrgLeader searchOrgLeader(int o_id)</p> |
| Event | <p>This is the entity class event, which contains the data about the event.</p> <p><u>Attributes:</u></p> |

| | |
|--------------|---|
| | <pre> private int eventID private String title private String dateStart private String dateEnd private String timeStart private String timeEnd private String type private String orgAcronym private String venue private String scope private String description private int limit <u>Methods:</u> private void setEventID(int theID) private void setTitle(String title) private void setDescription(String description) private void setDate(String dateStart, String dateEnd) private void setTime(String timeStart, String timeEnd) private void setvenue(String venue) private void setType(String type) private void setlimit(int lim) </pre> |
| PCLEvent | <p>This is the entity class list of events, which contains all of the events created by an org leader.</p> <p><u>Attributes:</u></p> <pre>private Event[] events</pre> <p><u>Methods:</u></p> <pre> private void addEvent(Event e) private Event searchEvent(int e_id) private void deleteEvent(Event e) </pre> |
| Announcement | <p>This is the entity class announcement, which contains the data about the announcement.</p> <p><u>Attributes:</u></p> <pre> private int eventID private int annID private String orgAcronym private String body private String dateAndTimeCreated privateString SN_col </pre> |

| | |
|--------------------|--|
| | <p><u>Methods:</u></p> <pre>private void seteventID(int e_id) private void setannID(int a_id) private void setbody(String bod)</pre> |
| PCLAnnouncement | <p>This is the entity class list of announcements, which contains all of the announcement created by an org leader.</p> <p><u>Attributes:</u></p> <pre>private Announcement[] announcements</pre> <p><u>Methods:</u></p> <pre>private void addAnn(Announcement a) private Announcement searchAnn(int a_id) private void deleteBookmark(Bookmark b)</pre> |
| StudentBookmark | <p>This is the entity class student bookmark, which contains the data about the events bookmarked by the student.</p> <p><u>Attributes:</u></p> <pre>private <ArrayList>(Student); private <ArrayList>(Event);</pre> <p><u>Methods:</u></p> |
| PCLStudentBookmark | <p>This is the entity class list of student bookmark, which contains all of the events bookmarked by a student.</p> <p><u>Attributes:</u></p> <pre>private Bookmark[] bookmarks</pre> <p><u>Methods:</u></p> <pre>private void addBookmark(Bookmark b) private Bookmark searchBookmark(int s_id, int e_id) private void deleteBookmark(Bookmark b)</pre> |
| StudentEvent | <p>This is the entity class student event, which contains the data about the events joined by the student.</p> <p><u>Attributes:</u></p> <pre>private <ArrayList>(Student); private <ArrayList>(Event);</pre> <p><u>Methods:</u></p> |
| PCLStudentEvent | <p>This is the entity class list of events, which contains all of the events joined by a student.</p> <p><u>Attributes:</u></p> <pre>private Event[] events</pre> <p><u>Methods:</u></p> <pre>private void addEvent(Event b) private Event searchEvent(int s_id, int e_id)</pre> |

| | |
|--|-----------------------------------|
| | private void deleteEvent(Event b) |
|--|-----------------------------------|

Data Sources Package:

| File Name or Database Name | Description |
|----------------------------|--|
| IskolendarDatabase | This is the data source that contains the information that will be displayed on the system's calendars and information about the users of the system found in a relational database system. It contains the tables Student, OrgLeader, Event, Announcement, StudentEvent, and StudentBookmark. |