DCS Codex Mobile App
Data Design Document

Submitted to:

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

Submitted by:
Borja, Kim
Pilipina, Jigger Angelo
Valencia, Ian Benedict

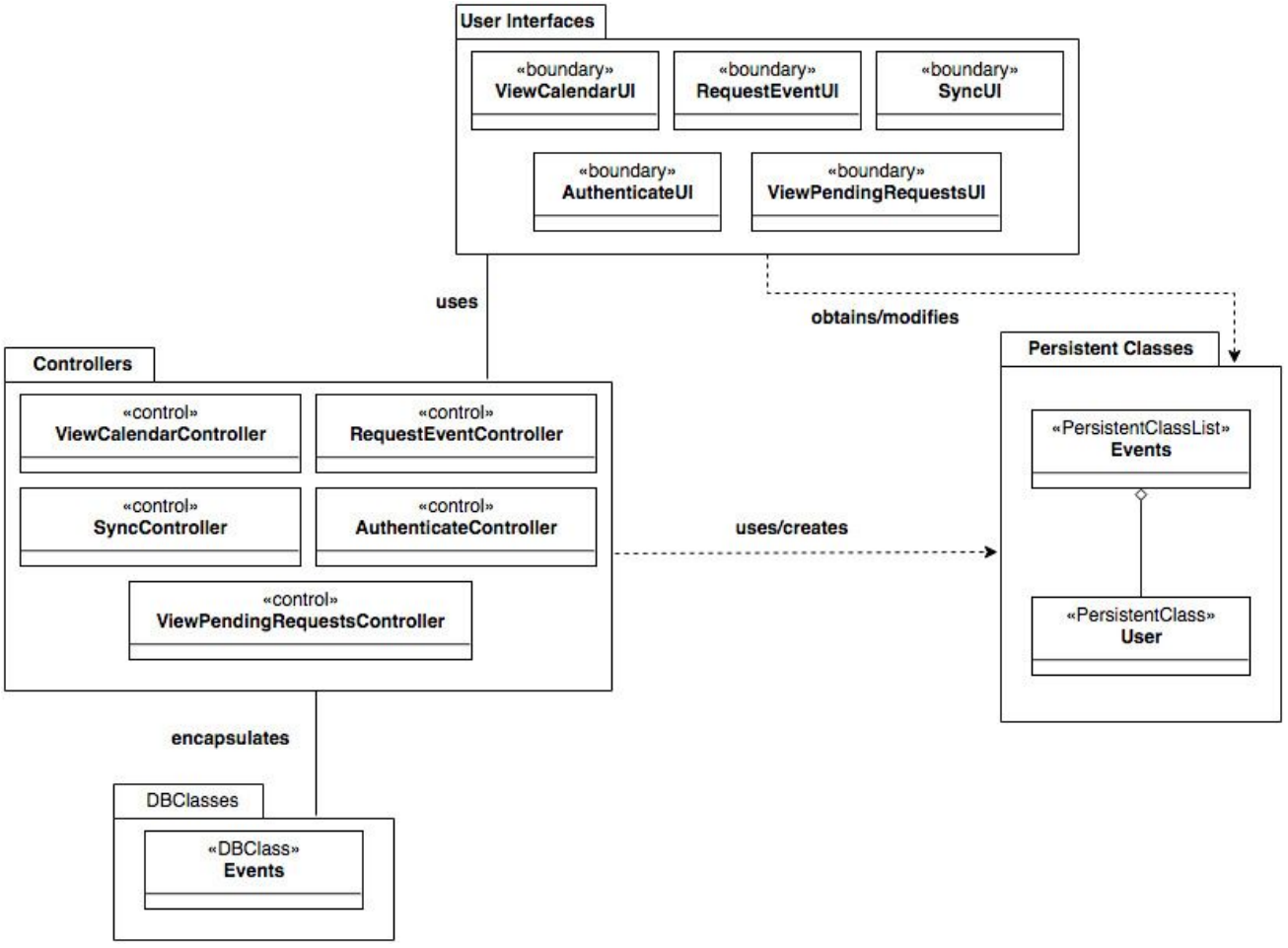
In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2017-2018

Revision Control

History Revision:

Revision Date	***Person Responsible***	***Version Number***	***Modification***
11/22/17	Borja, Kim	1.0	Initial Document;
11/22	Valencia, Ian	2.0	Added sample tables from database

Data Design:


User Interfaces
«boundary»
ViewCalendarUI
«boundary»
RequestEventUI
«boundary»
SyncUI
«boundary»
AuthenticateUI
«boundary»
ViewPendingRequestsUI
uses
obtains/modifies
Controllers
Persistent Classes
«control»
ViewCalendarController
«control»
RequestEventController
«control»
SyncController
«control»
AuthenticateController
«control»
ViewPendingRequestsController
uses/creates
«PersistentClassList»
Events
«PersistentClass»
User
encapsulates
DBClasses
«DBClass»
Events

Data Access Object (DAO) Classes:

Class Name	Description
ViewCalendarController	This control class enables the student to select a date among those shown in the Calendar of the current month in order to view events related to that date.
RequestEventController	This control class enables the student to request an event to the administrator.
SyncController	This control class enabled the student to view his/her pending requests and the status of each.
AuthenticateController	This control class enables the student to send unsent event requests to the database.
ViewPendingRequestsController	This control class enables the student to enter his/her student number which will then serve as student ID for whoever is using the mobile application.

TransferObject Classes:

Class Name	Description
Events	Event-handling
User	User-handling

List of Data Source:

Data Source Name: DBDCSCodex
Description: DBDCSCodex is a relational database containing 2 tables, User, and Events. The User table has 2 columns. The Student_id column contains the student number of the user while the Surname column contains the user's surname. The Events table has 7 columns. The Event_id column contains a unique id for each event requested. EventName column contains the name of the event. The Subject column contains which subject the event is under and the Professor column contains the professor/instructor related to the event. The Date column contains the date of deadline and the Time column contains the time of the deadline date. Lastly, the Sender column indicates which user has requested the event.

Sample Tables from the Database:

User

Student_id	Surname
INT	VARCHAR(100)
999999999	X(100)
PK, UA	**NN, UA**
201512345	Cruz
201402695	Dela Cruz
201396424	Delacruz

Events

Event_id	EventName	Subject	Professor	Date	Time	Sender
SMALLINT	VARCHAR(100)	CHAR(100)	VARCHAR(100)	DATE	TIME	INT
9999	X(100)	X(100)	X(100)	Mon dd, yyyy	hh:mm	999999999
PK	**NN, UA**	**NN, UA**	**NN, UA**	**NN, UA**	**NN, UA**	**NN, FK**
0001	MP1 Deadline	CS 140	Sir Juancho	Dec 12, 2017	23:59	201512345
0002	Workshop 21 Deadline	CS 191	Ma'am Weng	Dec 01, 2017	17:00	201402695
0003	MP2 Deadline	CS 11	Sir Tope	Dec 03, 2017	23:59	201396424