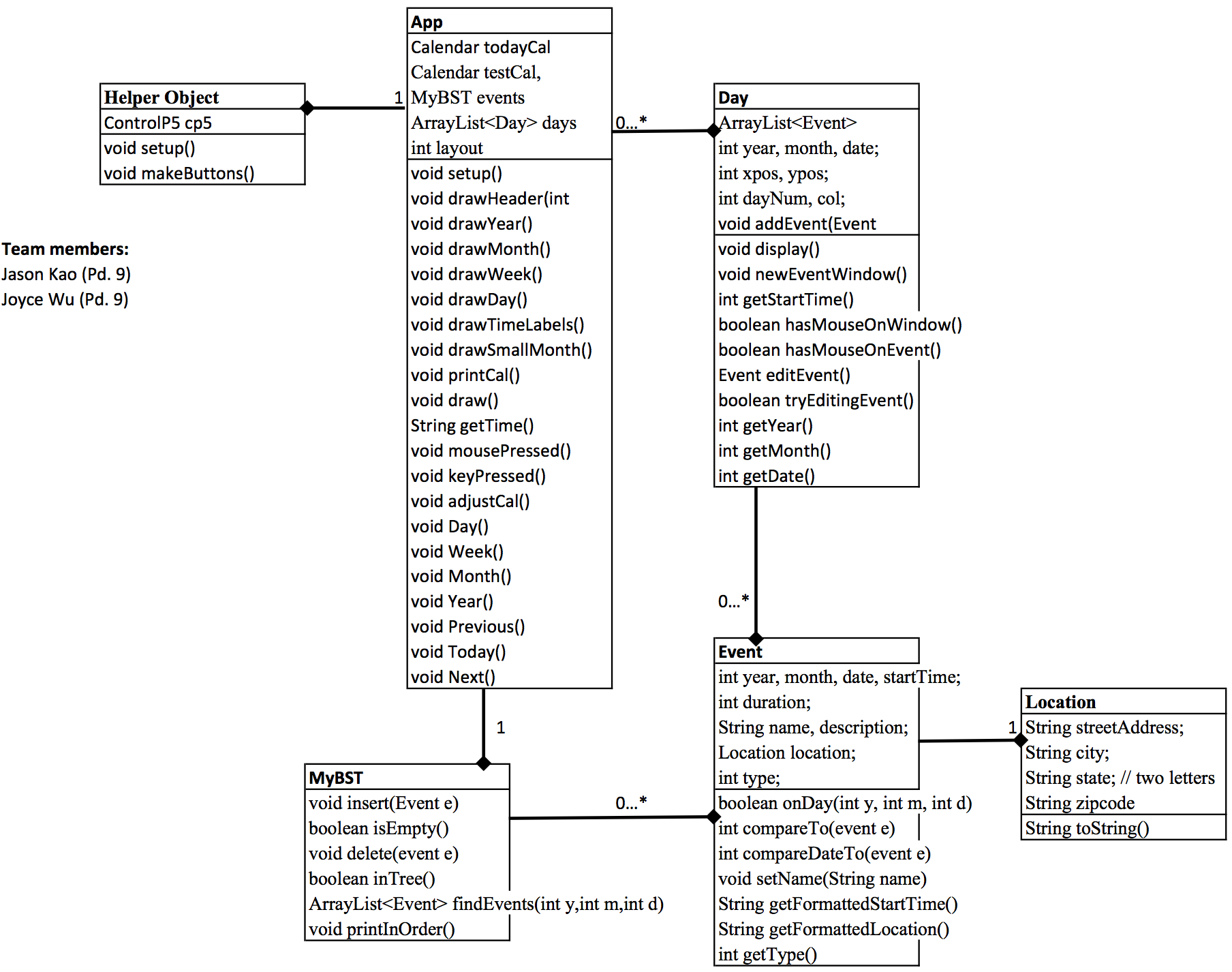
**Team Members Period 9**

Jason Kao

Joyce Wu

**BUNGA**



1. BRIEF SUMMARY: Calendar inspired by Google calendars to keep track of events throughout the day, week, month, and year. The basic features are listed below for all layouts (day, week, month, year)
2. FEATURES
   1. All layouts
      1. add any number of events on any number of days by clicking on the box a day belongs to.
      2. modify events’ name, date, duration, description, and location by clicking on an event box
      3. use Previous, Next, and Today buttons to determine which interval of days to show
      4. a red circle will show which date is the current one
   2. Week Layout
      1. display events not only by date but by time
      2. red time line shows the current time
   3. Day Layout
      1. display events by time, has red time line
      2. small mini calendar in the left sidebar shows the date with perspective to its entire month
      3. sidebar shows specific details about the event last modified (by default it shows the earliest event of the day)
   4. Year Layout
      1. no event modification ability
      2. shows the entire year with twelve blocks representing the months
3. DEVELOPMENT STAGES
   1. Skeleton with minimal functionality
      1. Basic Processing
      2. Creating a basic layout for all layouts
      3. Create Objects without reading/writing to files
   2. Major features
      1. data management and file writing
      2. creating events based on mouse click
      3. editing events with mouse click
   3. Making it pretty!
      1. setting a color scheme
      2. simple button and layout changes to make the App more aesthetically pleasing
   4. Things we did not get to
      1. Drag & drop event blocks
      2. sending email reminders

App Object

Calendar testCal, todayCal

MyBST events

ArrayList<Day> days

int layout

void setup() sets up calendar

void drawHeader(int layout) sets up header according to layout

void drawYear() displays calendar for entire year

void drawMonth() displays month calendar

void drawWeek() displays week calendar

void drawDay() displays day calendar

void drawTimeLabels() draws time labels on the side of Week & Day layout

void drawSmallMonth() draws the small month seen in year and day layout

void printCal() is a convenient method for printing the testCalendar

void draw()

String getTime() retrieves current time in hh:mm(a/p) format

void mousePressed()

void keyPressed()

void adjustCal()

void Day() draws day

void Week() draws week

void Month() draws month

void Year() draws year

void Previous() previous time increment

void Today() centers the time increment on the present day

void Next() next time increment

Day Object

ArrayList<Event> todayEvents holds a list of events for the day

int year, month, date;

int xpos, ypos;

int dayNum, col;

void addEvent(Event e) adds an event to the day

void display() displays the day with respect to layout

void newEventWindow() creates a New Event window

int getStartTime() gets start time based on mouse position

boolean hasMouseOnWindow()

boolean hasMouseOnEvent()

Event editEvent()

boolean tryEditingEvent()

accessor methods for year, month, date

Event Object

int year, month, date, startTime;

int duration;

String name, description;

Location location;

int type;

boolean onDay(int y, int m, int d)

int compareTo(event e)

int compareDateTo(event e)

void setName(String name)

String getFormattedStartTime()

String getFormattedLocation()

int getType()

Helper Object

ControlP5 cp5

void setup()

void makeButtons() creates buttons on App

Location Object

String streetAddress;

String city;

String state; // two letters

String zipcode

MyBST Object

void insert(Event e)

boolean isEmpty()

void delete(event e)

boolean inTree()

ArrayList<Event>findEvents(int year, int month, int date)

void printInOrder() prints the events in order