Discipline of ICT
School of Technology, Environments and Design
College of Sciences and Engineering

UTAS

UNIVERSITY OF TASMANIA

Group Assignment Cover Sheet

Unit Code: KIT506 Unit Name: Software Design and Development and software Application Design and Implementation
Tutorial Day/Time: Monday 3pm-5pm (of one group member)
Group Name/#: k506h_al_m1500_5 Assignment Title/#: Assignment 1-OO Design

We declare that all material in this assignment is our own work except where there is clear acknowledgement or reference to the work of others and we have complied and agreed to the University statement on Plagiarism and Academic Integrity on the University website at http://www.utas.edu.au/plagiarism or in the Student Information Handbook. We are aware that our assignment may be submitted to plagiarism detection software, and might be retained on its database.

Family Name: Wang Given Name: Haijan
Student ID: 480898 User Name: hwang16 @postoffice.utas.edu.au
Student Signature: 王海珅 Date: 28/8/2018

Family Name: YAO Given Name: XIXI
Student ID: 423488 User Name: xixiy @postoffice.utas.edu.au
Student Signature: Date: 28/08/2018

Family Name: Liu Given Name: Yuming
Student ID: 478673 User Name: yliu68 @postoffice.utas.edu.au
Student Signature: Yuming Liu Date: 28/08/2018

Family Name: Given Name:
Student ID: User Name: @postoffice.utas.edu.au
Student Signature: Date:

Family Name: Given Name:
Student ID: User Name: @postoffice.utas.edu.au
Student Signature: Date:

If more than 5 people in the group, use two coversheets

Statement on Plagiarism and Academic Integrity

Plagiarism is a form of cheating. It is taking and using someone else's thoughts, writings or inventions and representing them as your own, for example:

- using an author's words without putting them in quotation marks and citing the source;
- using an author's ideas without proper acknowledgment and citation; or
- copying another student's work.

If you have any doubts about how to refer to the work of others in your assignments, please consult your lecturer or tutor for relevant referencing guidelines, and the academic integrity resources on the web at http://www.academicintegrity.utas.edu.au.
The intentional copying of someone else's work as one's own is a serious offence punishable by penalties that may range from a fine or deduction/cancellation of marks and, in the most serious of cases, to exclusion from a unit, a course or the University. Details of penalties that can be imposed are available in the Ordinance of Student Discipline – Part 3 Academic Misconduct, see http://www.utas.edu.au/__data/assets/pdf_file/0006/23991/Ordinance-9-Student-Discipline-December-2017.pdf
The University reserves the right to submit assignments to plagiarism detection software, and might then retain a copy of the assignment on its database for the purpose of future plagiarism checking.

Contents

• Part 1: Scenarios

1.1. Use Case #8: User_views_StaffList

Requirement:

The user shall be able to view the list of employees working in the school at present.

Overview:

When the application was launched, the view shows a tab on which the user can click to view the list of current staff in the University. The list should show the Names of the staffs in the form of "Given Name, Family Name (Title)" in alphabetic order. Additionally, the list should be filtered with some conditions to let the user to search specific staffs.

Preconditions:

1. The application is started successfully.

Scenario:

Action	Software Reaction
1. User open the HRIS system	1. Open GUI 2. Create MainView by default 3. Show Mainview to user
2. User clicks the tab named "Staff" in MainView	1. Open the database 2. Get staff data from database 3. Sort the data in alphabetic order 4. Cache staff data of list 5. Set data in the StaffListView 6. Show StaffListView in MainView
3. User select staff category in the combo box	1. Get staff data of list from cache 2. Select staff belonging to the specified category 3. Update the table of StaffListView
4. User types search pattern in the filter text box	1. Get staff data of list from cache 2. Select staff data of list by filter 3. Update the table of StaffListView
5. User clicks "Home" button	1. Return to MainView

Scenario Notes:

The user normally can click the Staff tab to view the staff list, which is Action 1 and 5, Action 5 is used to exit the user case. Optionally the user can filter the staff list with specified criteria, by either selecting category or typing pattern in the filter text box, which are done through Action 2 and 3. The user also can view the details of a staff through Action 4.

Post Conditions

Staff list is shown in the Staff View

Required Views: (GUIs)

MainView

StaffListView

Exceptions:

1. The network is down leading to connecting to database failed, inform user
2. Database space is full, inform user

Use Cases utilised

UC16_User_selects_StaffDetails

1.2. Use Case #16: User_selects_StaffDetails

Requirement:

When the user selects a name in the StaffListView the system will show more details about the staff member (referred to as the Staff Details view), which should include: Name; Campus; Phone Number; Room Location; Email Address; Photo; Consultation hours; Table of units he or she is involved with in the current semester.

Overview:

1. When the user selects a staff's name in the list, the system will shift to a new window to show more details about the staff member:
    - Name
    - Campus
    - Phone Number
    - Room Location
    - Email Address
    - Photo
    - Consultation hours
    - Table of units he or she is involved with in the current semester

2. When the user selects a unit code in the table of units, the system will pop up a box to show the timetable view that each staff member's current availability can be displayed:
    - 'teaching' (with details of the unit code and room) - if they are in a timetabled class;
    - 'consulting' - if it is during their consultation times;
    - 'free' (can just show blank) - otherwise.

Preconditions:

The list of staff (StaffListView) has displayed.

Scenario:

Action	Software Reaction
1. User clicks a staff's name in the StaffListView	1. Get staff detail data from database. 2. Sort the data as requirement. 3. Shift to a new window to show more details about the staff member (StaffDetailsView):

	• Name • Campus • Phone Number • Room Location • Email Address • Photo • Consultation hours • Table of units he or she is involved with in the current semester
2. User clicks “Home” button in StaffDetailsView	1. Return to the MainView.
3. User clicks a unit code in the table of units in StaffDetailsView	1. Get staff timetable of this unit from database. 2. Sort the data as requirement. 3. Shift page to show the timetable view that each staff member’s current availability can be displayed (StaffTimetableView): • ‘teaching’ (with details of the unit code and room) - if they are in a timetabled class; • ‘consulting’ - if it is during their consultation times; • ‘free’ - otherwise.
4. User clicks “Home” button in StaffTimetableView	1. Return to the MainView.

Scenario Notes:

The typical path is actions 1 then 2, or actions 1 then 3 then 4. If there are any errors clicking operation, the system will display the previous page. E.g. if there is an error with action 1, the system will stay in StaffListView. If there is an error with action 3, the system will stay in the StaffDetailsView.

Post Conditions

1. Contents of selected staff’s details displayed in StaffDetailsView
2. Current availability of selected staff’s unit code displayed in StaffTimetableView

Required Views: (GUIs)

StaffListView
StaffDetailsView
StaffTimetableView

Exceptions:

1. There is no details data about the selected staff; inform user and stay in the current view.
2. There is no timetable data with selected staff member; inform user

Use Cases utilised

UC8_User_views_StaffList (precondition)
UC19_User_shows_ActivityGrid

1.3. Use Case #19: User_shows_ActivityGrid

Requirement:

It would enhance the Staff Details view if the staff member's activity (classes and consultation times) across a week could be displayed in a colour-coded grid.

Overview:

1. This grid should be toggled (displayed or hidden) via a button on the Staff Detail view.
2. The grid should have days of the week (Monday through Friday) as columns and hours of the day (9am until 4pm) as rows, with each cell's colour indicating the kind of activity at that time, but no other details shown.
3. Free time should be shown in white, while teaching and consultation times should be shaded in distinct colours that are distinguishable by those with common forms of colour blindness.

Preconditions:

The details of staff (StaffDetailsView) has displayed.

Scenario:

Action	Software Reaction
1. At the end of the StaffDetailsView, there is a button named "Show staff activity grid". User clicks this button.	1. Get staff activity data from database. 2. Sort the data as requirement. 3. Expand a grid under the button. The grid is displayed with the format and content are as follows (StaffActivityGrip): • The grid has days of the week (Monday through Friday) as columns and hours of the day (9am until 4pm) as rows. • Each cell's colour of the grid indicates the kind of activity at that time, but no other details shown. (Red and yellow are easier to distinguish by colour blind users.) E.g. Free time is shown in white, teaching time is shown in red and consultation time is shown in yellow. •At the right side of the grid, there is the direction about the colours' meaning. 2. Button name change from "Show staff activity grid" to "Hide staff activity grid".
2. User clicks this button "Hide staff activity grid".	1. Hide the staff activity grid. 2. Button name change from "Hide staff activity grid" to "Show staff activity grid".

Scenario Notes:

The typical path is action 1 then 2, or action 2 then 1. It depends on the status of the current page display. The default status is hiding the staff activity grip.

Post Conditions

The StaffActivityGrip displayed in StaffDetailsView.

Required Views: (GUIs)

StaffDetailsView

StaffActivityGrip

Exceptions:

1. If there is no activity with the staff, the system will display a message “There is no activity with this staff” under the button after action1, and button name change from “Show staff activity grid” to “Hide staff activity grid” (User can continue action 2).

2. If there is any error in downloading the staff activity data, the system will display a message under the button, requesting that the user try it again.

Use Cases utilised

UC16_User_selects_StaffDetails (precondition)

1.4. Use Case #23: User_views_UnitList

Requirement:

Generate a list of units under the control of the school. The format of the list is displayed in unit code, unit title, in alphabetical order, with a filter information bar.

Overview:

When you click on the unit list, a mesh table appears, divided into two columns with Uint code, Uint title. The contents of the table begin with the letter A and are arranged in alphabetical order. In the page, another filter search bar, when you enter Uint code or Uint title, it will list the matching entries.

Preconditions:

Under the school control, the Unit list option is listed below.>

Scenario:

Action	Software Reaction
1. User clicks the “Unit” button in MainView	1. Get unit data from database 2. Sort the data in alphabetic order 3. Cache staff data of list 4. Set data in the UnitListView 5. Show UnitListView • The column is presented as unit code and unit title.
2. User clicks on the filter bar and enter any unit code or unit title	1. Get unit data of list from cache 2. Select unit data of list by filter

	3. Update the table of UnitListView
3. Click on the home button	1. Return to the MainView.

Scenario Notes:

Only unit code or unit text information is displayed in the list.

Post Conditions

Can present lists and users can quickly filter valid information.

Required Views: (GUIs)

MainView

UnitListView

Exceptions:

Filter information cannot be automatically sorted alphabetically, in addition may be sorted by information match.

Use Cases utilised

UC24_User_selects_Unit

1.5. Use Case #24: User_selects_Unit

Requirement:

The user can click the unit code inside the table of units of that staff to view the timetable of the classes belonging to that unit.

Overview:

In the view of details of a staff, the user clicks a specific unit code in the table of units, the system then shows a window displaying the class timetables for that units.

Preconditions:

1. The application is launched successfully.
2. The view of the details of a specific staff has already been opened.

Scenario:

Action	Software Reaction
1. User clicks the unit code cell in the unit list	1. Get the list of class in the specified unit from database 2. Sort the list in chronological order 3. Construct UnitDetailsView for this unit 4. Populate table of the class view with the obtained data 5. Show the UnitDetailsView
2. User double click the item in the UnitDetailsView	1. Get the staff information from item just clicked 2. Shift to the StaffDetailsView for this staff

	(Go to UC16_User_selects_StaffDetails)
3. User clicks the Campus Tab on the top of UnitDetailsView	1. Get the campus name from the selected item in dropdown list of Unit Timetable View 2. Re-query database to get the classes located in the specified campus 3. Update UnitDetailsView
4. User click Home button	1. Return to the MainView.

Scenario Notes:

Typically, user perform Action 1 and 3 to just view the related class for the selected unit. Also, the staff Detail View in regarding with this staff can also be displayed if the user wants to view more information about this staff teaching this class by doing Action 2.

Post Conditions

The list of the class timetable for the chosen unit is shown.

Required Views: (GUIs)

UnitListView
UnitDetailsView
StaffDetailsView

Exceptions:

1. Database cannot be connected, inform the user
2. There is no available staff for the unit, maybe currently is still under negotiation by the faculty of University, inform the user about this.

Use Cases utilised

UC23_User_views_UnitList (precondition)
UC16_User_selects_StaffDetails
UC37_User_generates_ClashMap

1.6. Use Case #28: User_generates_HeatMap

Requirement:

Generate a heat map by clicking on the activity. There are two categories of heat maps, one is to generate a unit chart heat map, and the other is to generate a consultation time heat map.

Overview:

Generating a heat map expressed in color by a mesh, by selecting different categories, one is to generate a unit chart heat map, and the other is to generate a consultation time heat map. There is also a range of choices.

Preconditions:

It is necessary to count the data of the heat map through weekly activities.

Scenario:

Action	Software Reaction
1. Click on the heat map button in MainView	1. Get Heat map data of units for Hobart campus by default from database. 2. Sort the data as requirement. • Count the number of activities 3. Show HeatMapView in MainView • column is Monday through Friday, horizontal is from 9 am to 4 pm • with white indicating no events • with a solid colour indicating the greatest number of events • with a colour intermediate between white and the selected solid colour indicating Intermediate values (The primary colour of the heat map is red by default.)
2. Click on any of the campuses	1. Get Heat map data from database. 2. Sort the data as requirement. 3. Update HeatMapView in MainView
3. Click unit class or consulting time	1. Get Heat map data from database. 2. Sort the data as requirement. 3. Update HeatMapView in MainView
4. User selects one primary colour	1. Change the heat map colour. 2. Update HeatMapView in MainView
5. Click home button	1. Return to the MainView.

Scenario Notes:

The color gradient may not be obvious, so the number will be added to strengthen the representation in the grid

Post Conditions

Choose different categories to generate different ranges of content

Required Views: (GUIs)

MainView

HeatMapView

Exceptions:

Generate two heat maps, using the gradient of the grid and color to indicate the frequency of the event.

Use Cases utilised

None

1.7. Use Case #37: User_generates_ClashMap

Requirement:

The application should be able to generate clash map between units and consultation time of that staff who teaches this unit.

Overview:

The click the button in UnitDetailsView, then the system calculate the conflict time between lecture and consultation for the staff involved, then show this in a grid view.

Preconditions:

UnitListView and UnitDetailsView are already opened

Scenario:

Action	Software Reaction
1. User clicks Clash Map button in the UnitDetailsView	1. Get the timetable info for lectures pertaining to each unit 2. Get the timetable info for consultation for each unit 3. From each week day compare the lecture and consultation time, save the lectures and consultations with the same time and week day into a list to judge the clash time 4. Construct Clash Map Grid View, populate its cells with the saved clashed lecture and consultations 5. Show the ClashMapView
2. User clicks the Home button	1. Return to the MainView.

Scenario Notes:

User typically go through Action 1 and Action 2 to generate and clash map view

Post Conditions

The view showing clash between lecture and consultation for all unit is displayed

Required Views: (GUIs)

UnitDetailsView

ClashMapView

Exceptions:

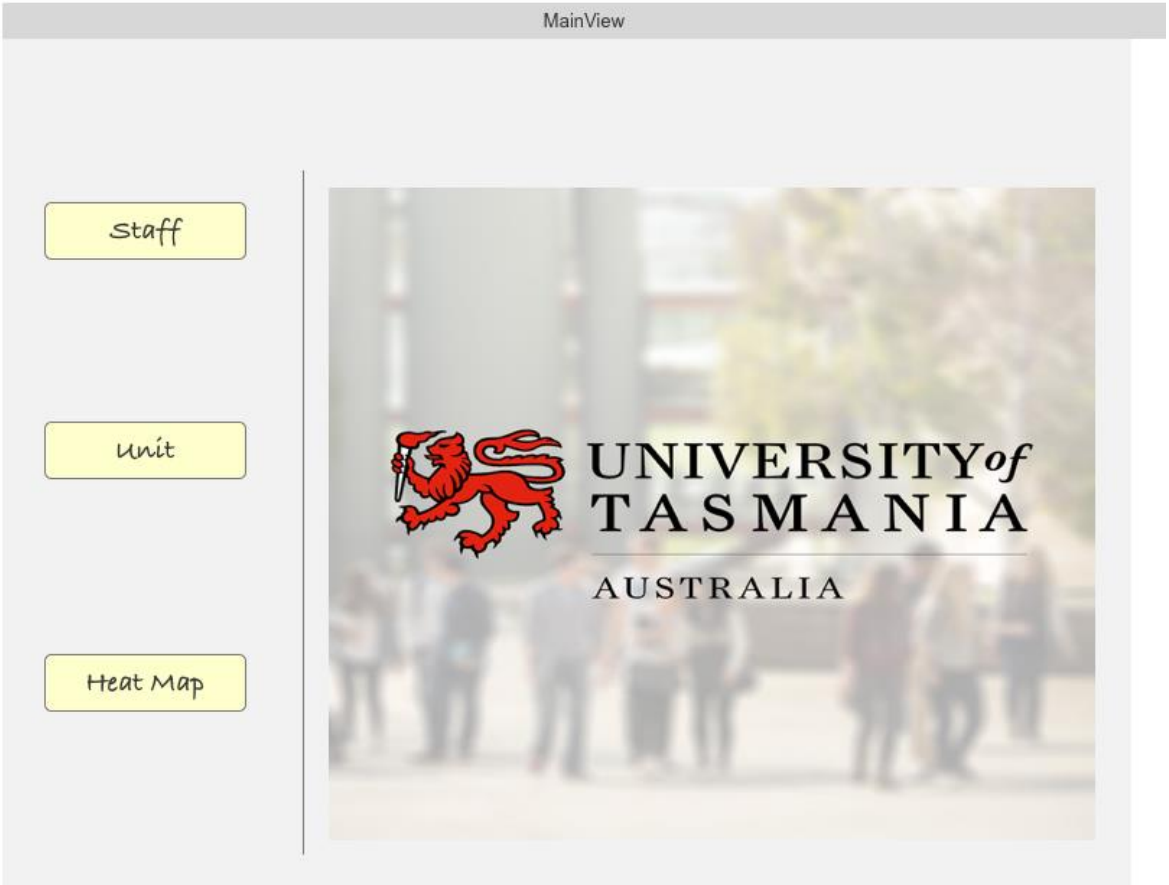
1. Too many class and consultations resulting in calculation crashed

Use Cases utilised

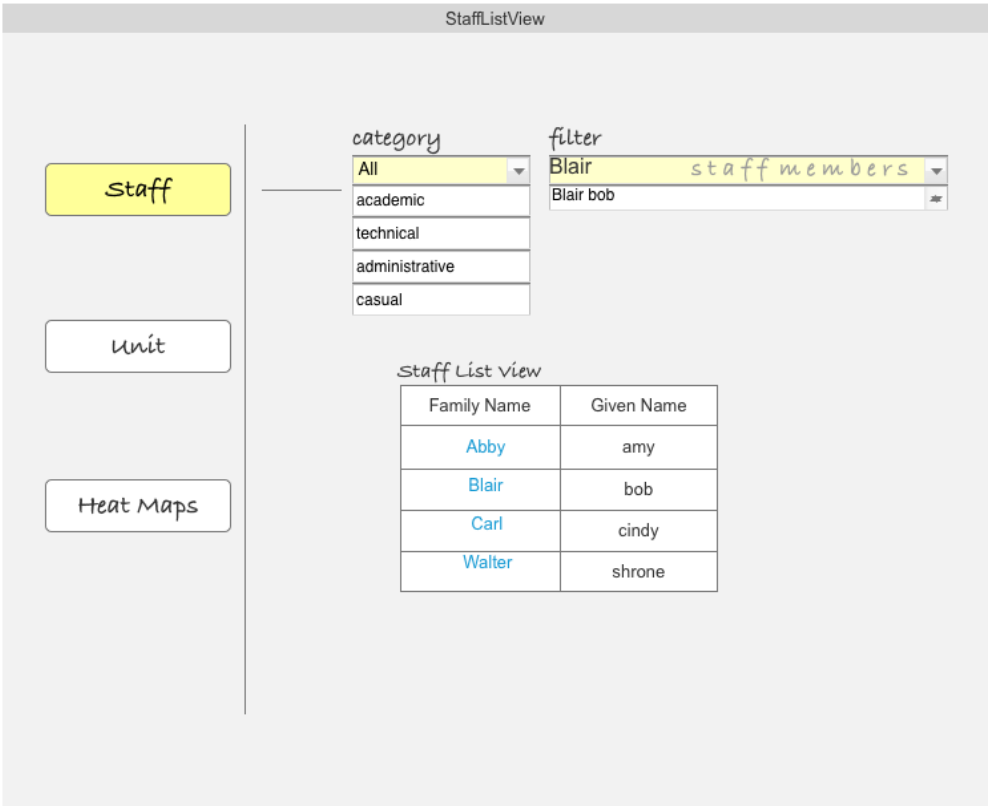
UC24_User_selects_Unit(precondition)

• Part 2: GUIs

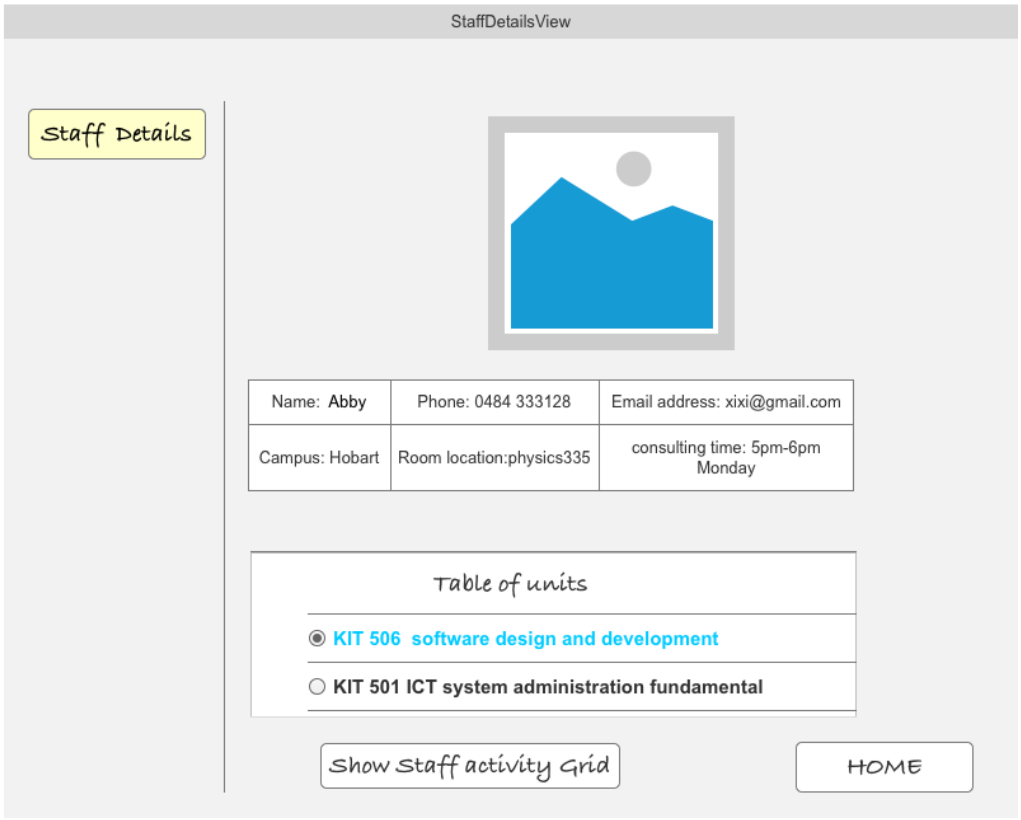
2.1. MainView



2.2. StaffListView



2.3. StaffDetailsView



2.4. StaffTimetableView

StaffTimetableview

Staff Timetable Box

Staff name: Abby

Unit: KIT 506 software design and development

	Monday	Tuesday	Wednesday	Thursday	Friday
9am-11am	teaching 506 physical 333	consulting	consulting	consulting	teaching 506 physical 333
11am-1pm	free	teaching 506 physical 333	consulting	teaching 506 physical 333	free
1pm-3pm	teaching 501 physical 376	free	teaching 506 physical 333	free	teaching 501 physical 376
3pm-5pm	consulting	teaching 501 physical 376	free	teaching 501 physical 376	consulting

HOME

2.5. StaffActivityGrid

StaffActivityGrid under StaffDetailsView

Hide Staff activity Grid

HOME

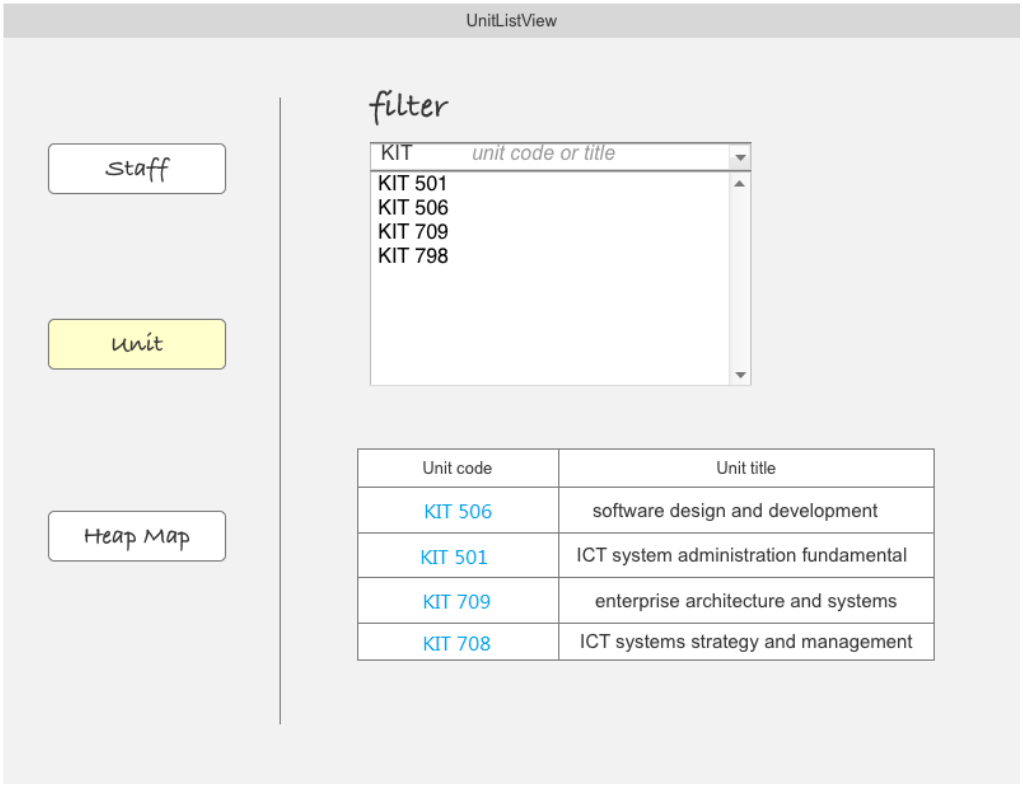
Activity Timetable

	9am-11am	11am-1pm	1pm-3pm	3pm-5pm
Monday				
Tuesday				
Wednesday				
Thursday				
Friday				

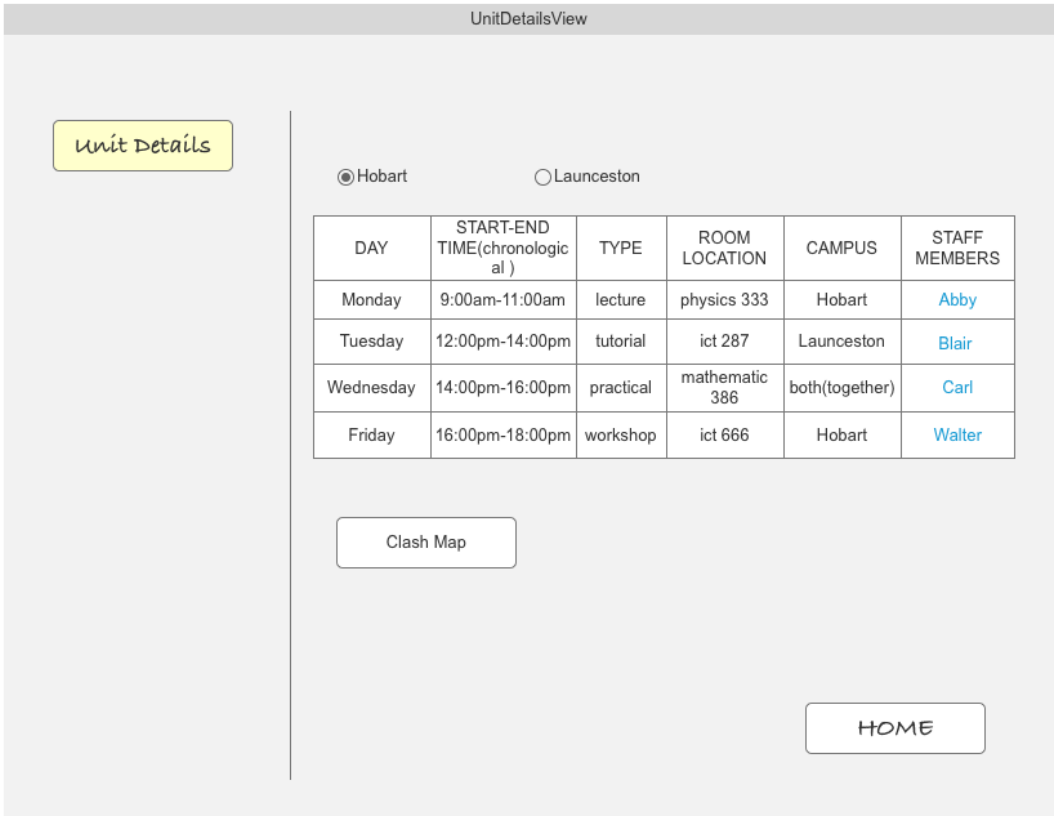
Teaching : blue in grid

consulting : yellow in grid

2.6. UnitListView

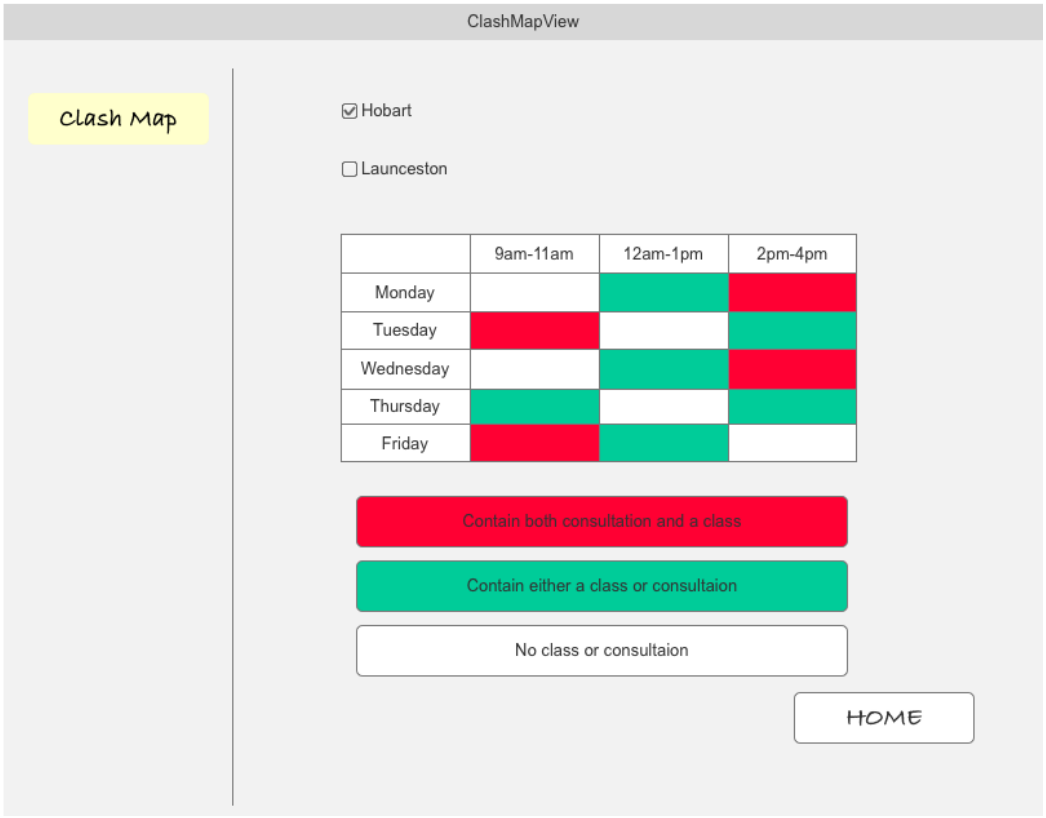


2.7. UnitDetailsView

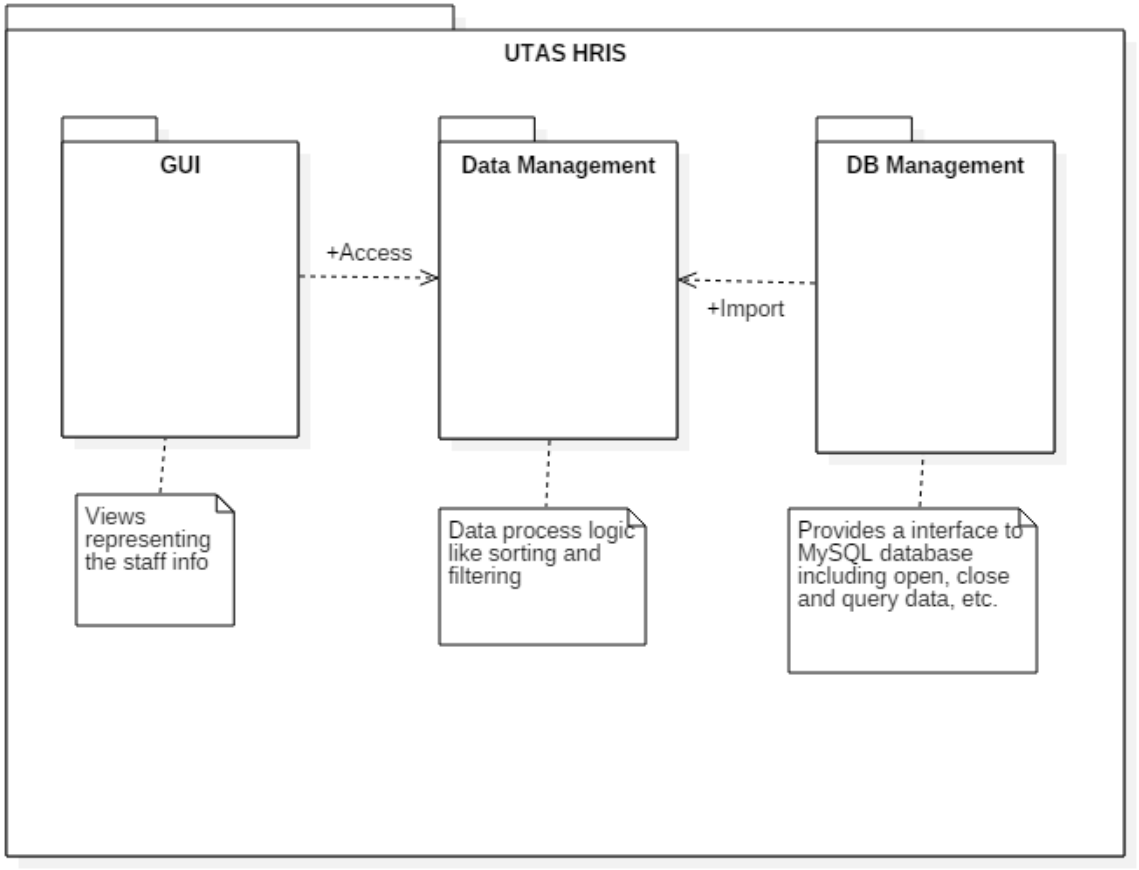


2.8. HeatMapView

2.9. ClashMapView

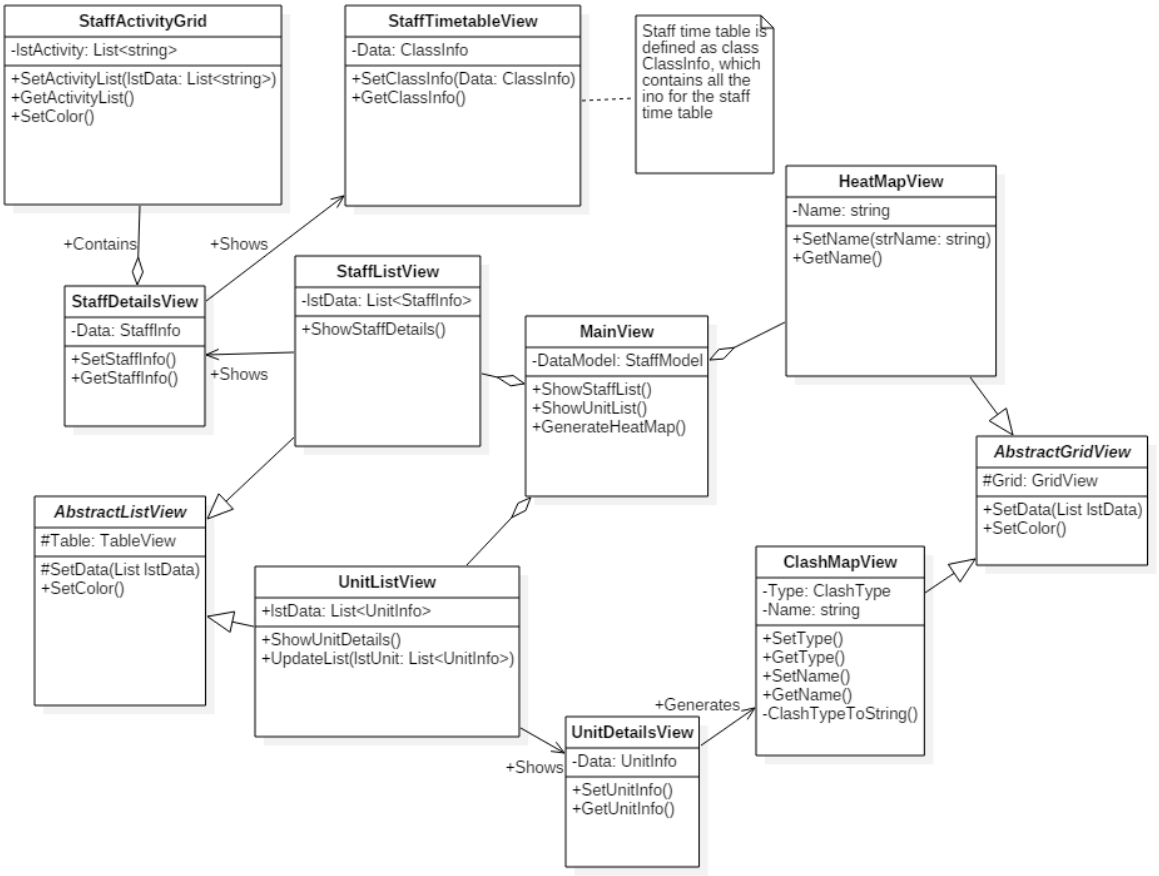


- **Part 3: Package diagram**

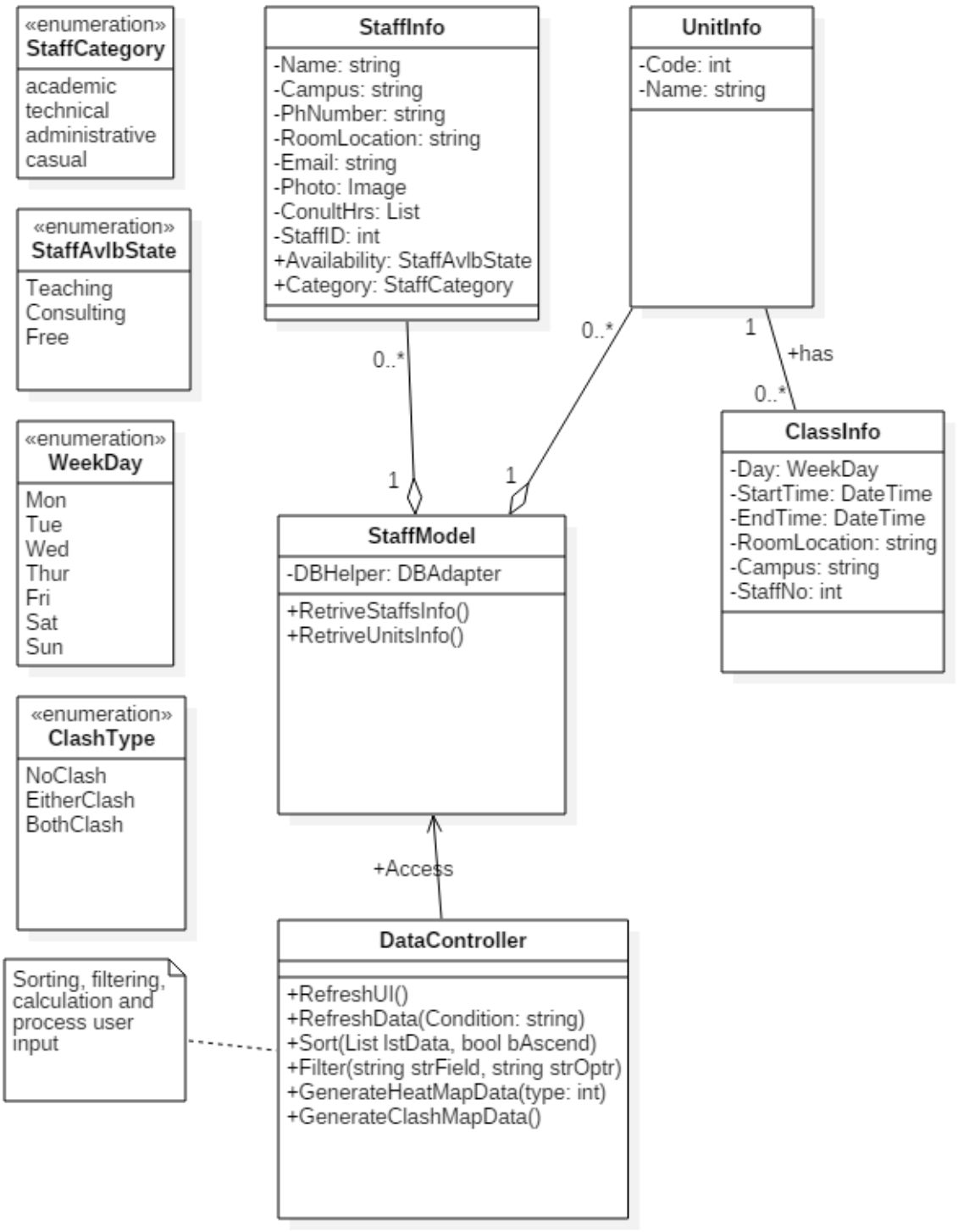


• Part 4: Class diagrams

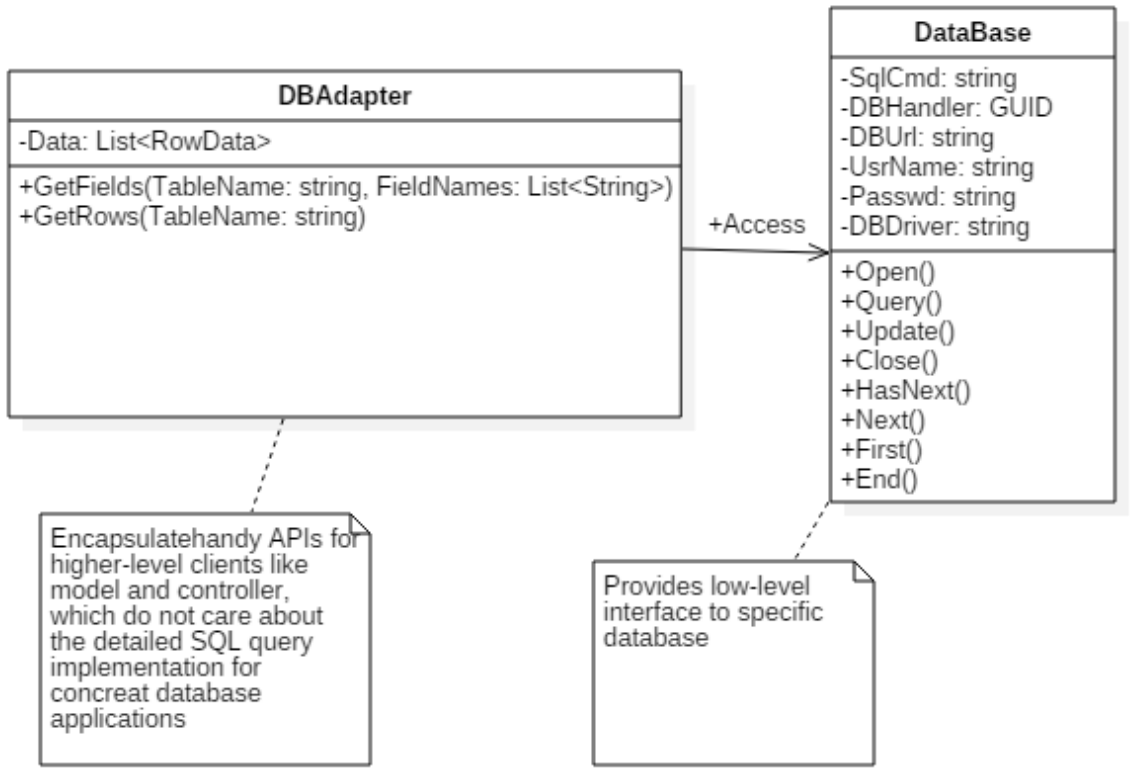
4.1. Class diagram in “GUI” Package



4.2. Class diagram in "Data Management" Package

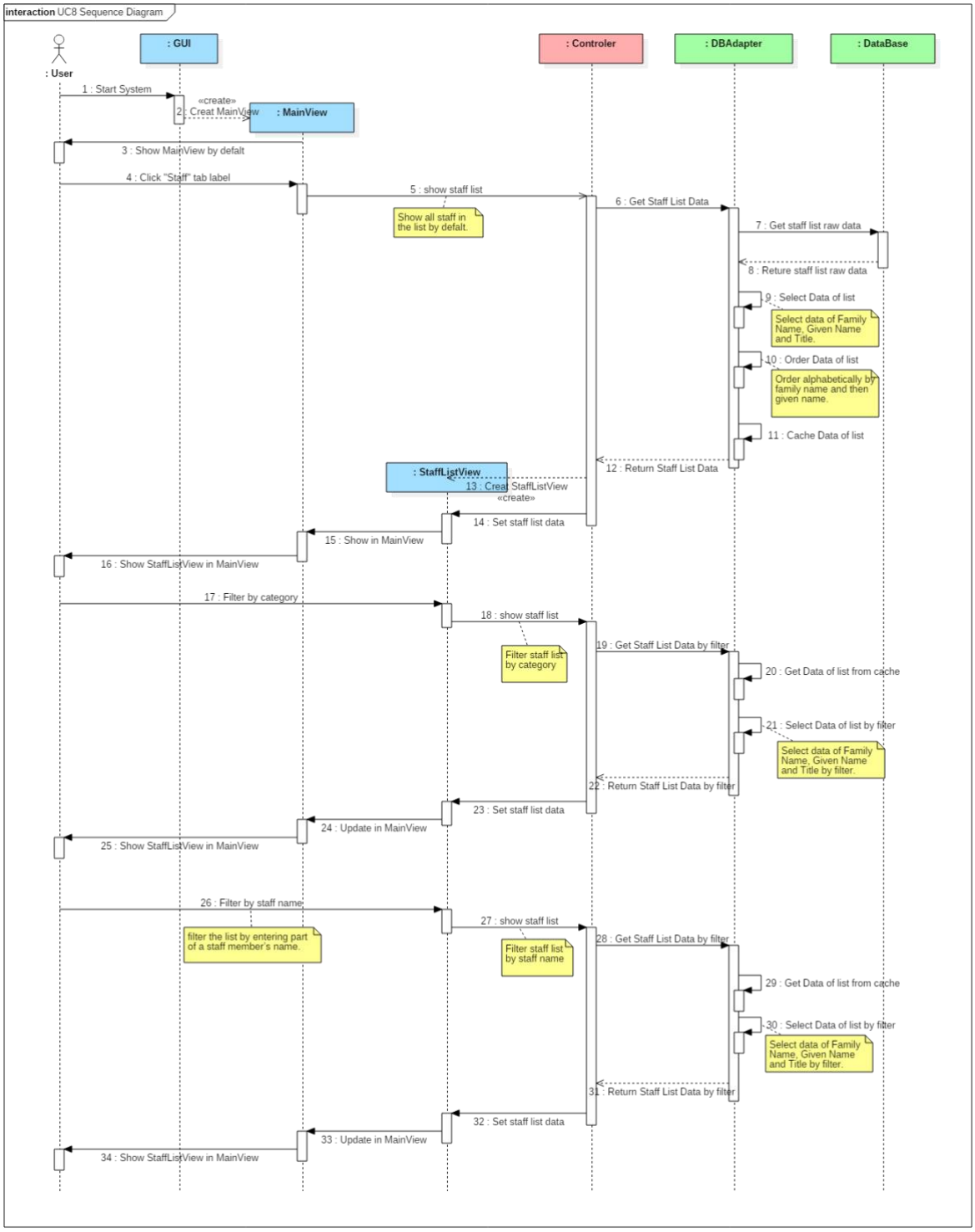


4.3. Class diagram in “DB Management” Package

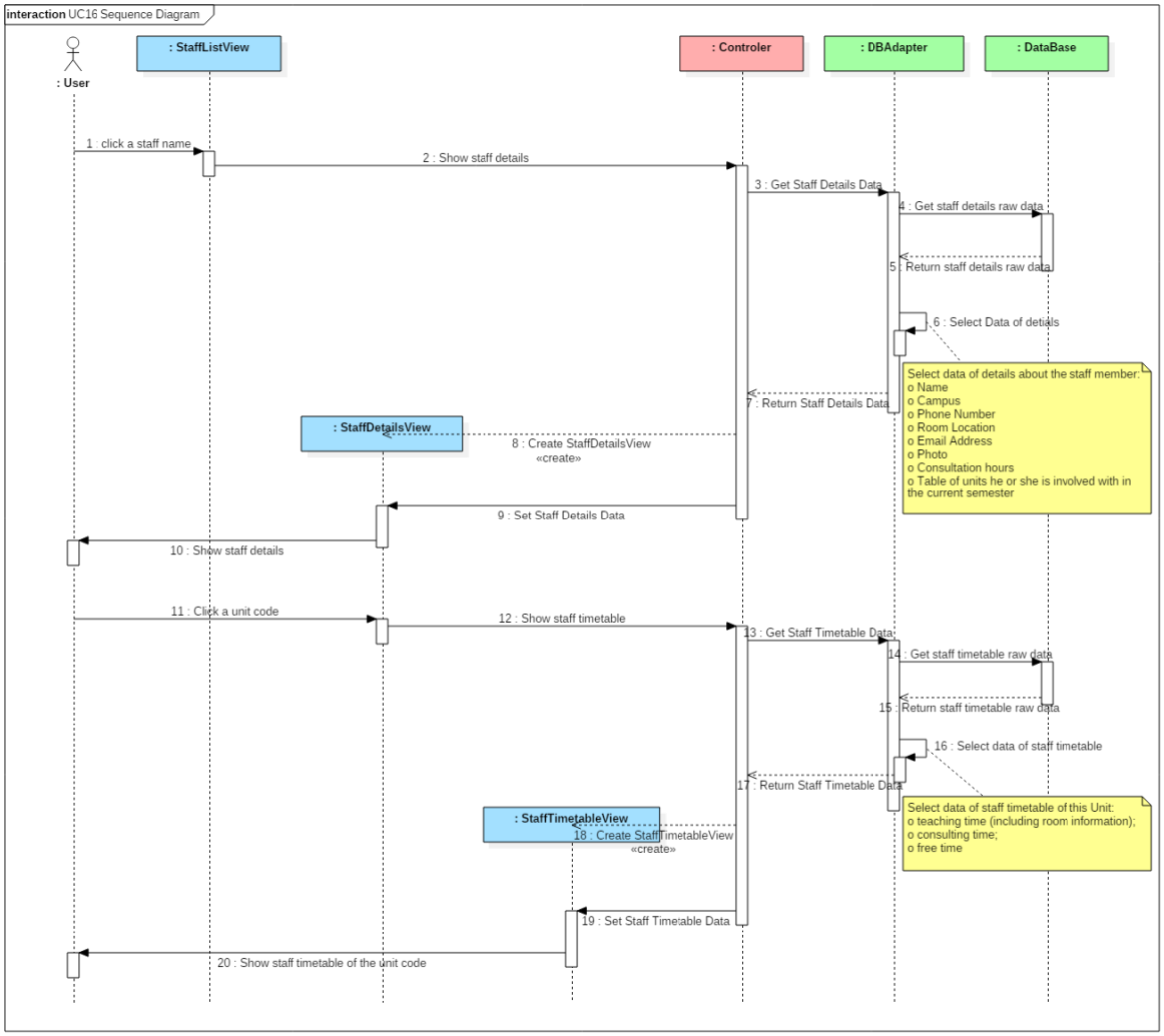


• Part 5: Sequence diagrams

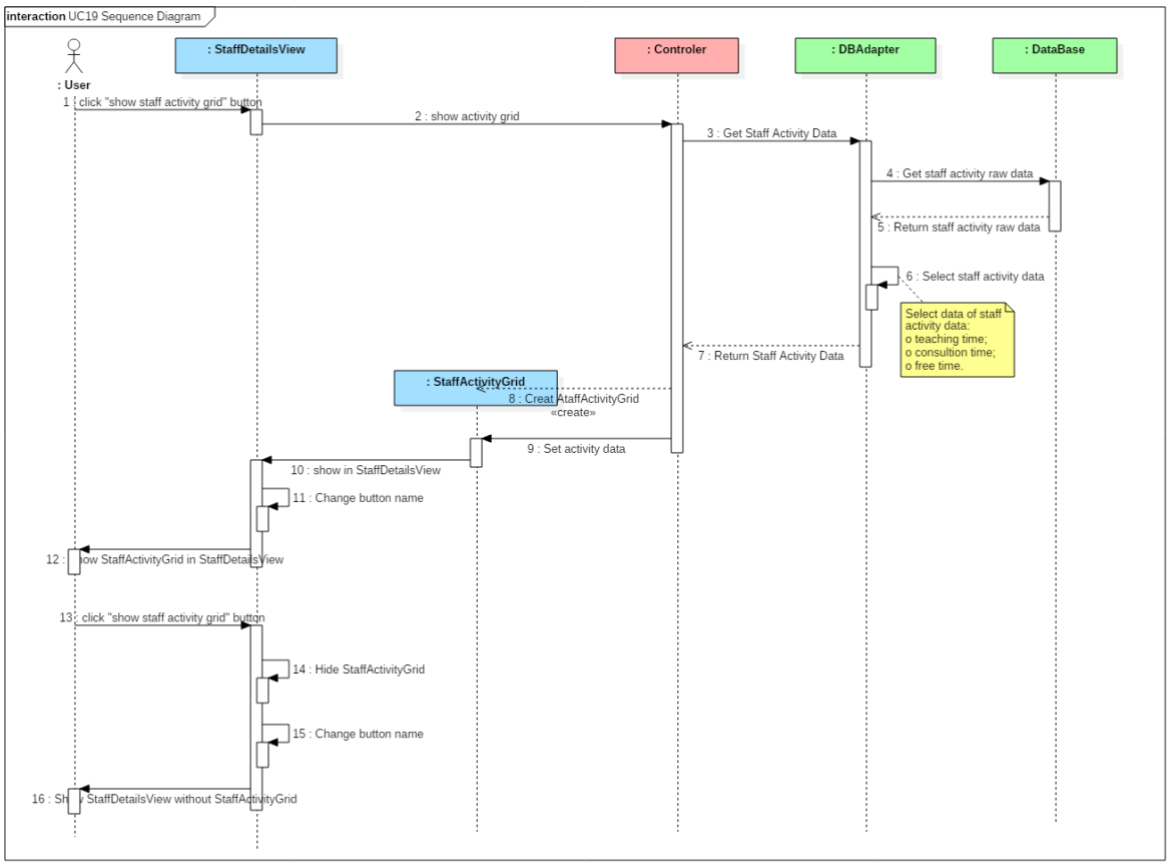
5.1. Sequence diagram of UC8_User_views_StaffList



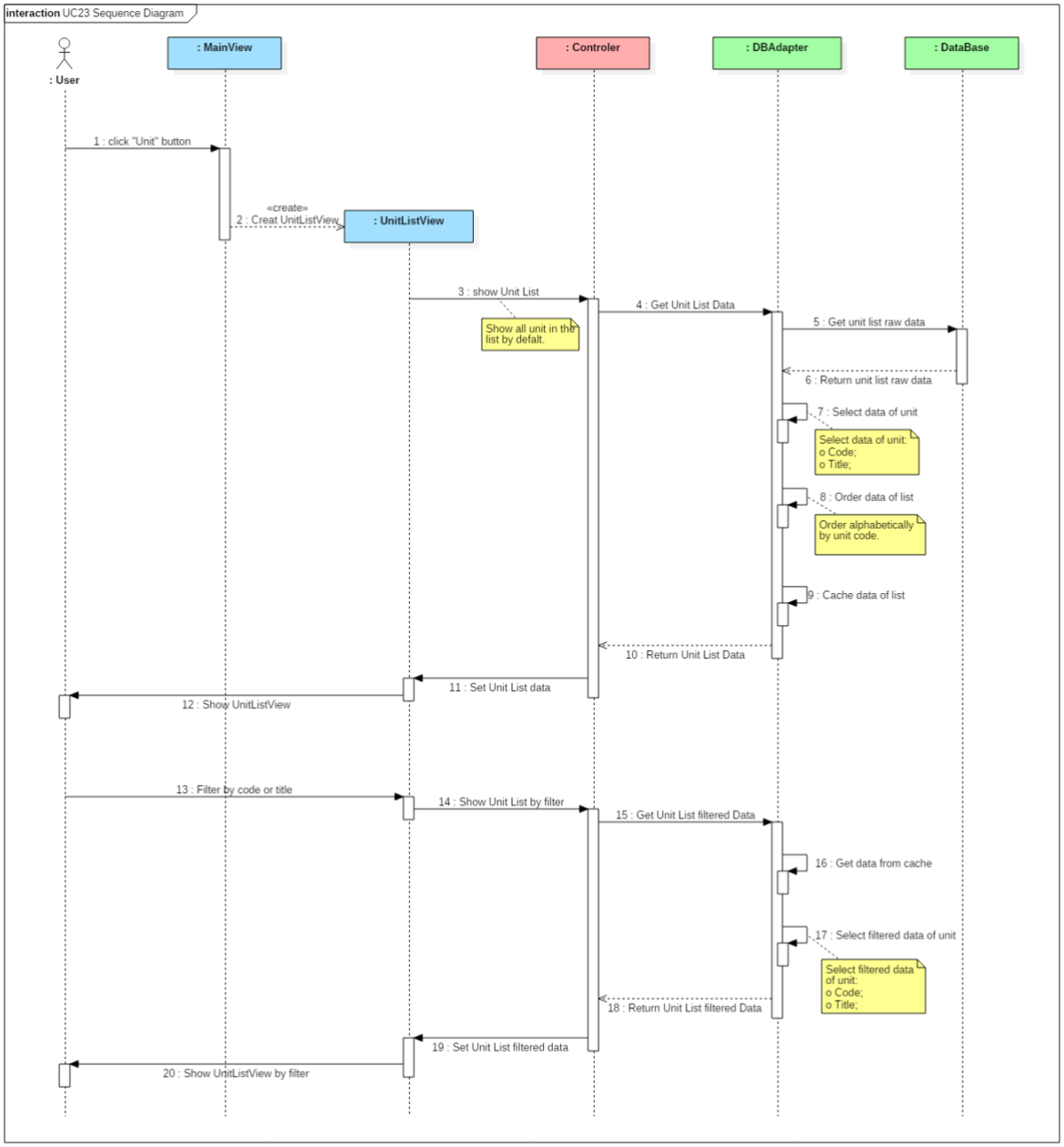
5.2. Sequence diagram of UC16_User_selects_StaffDetails



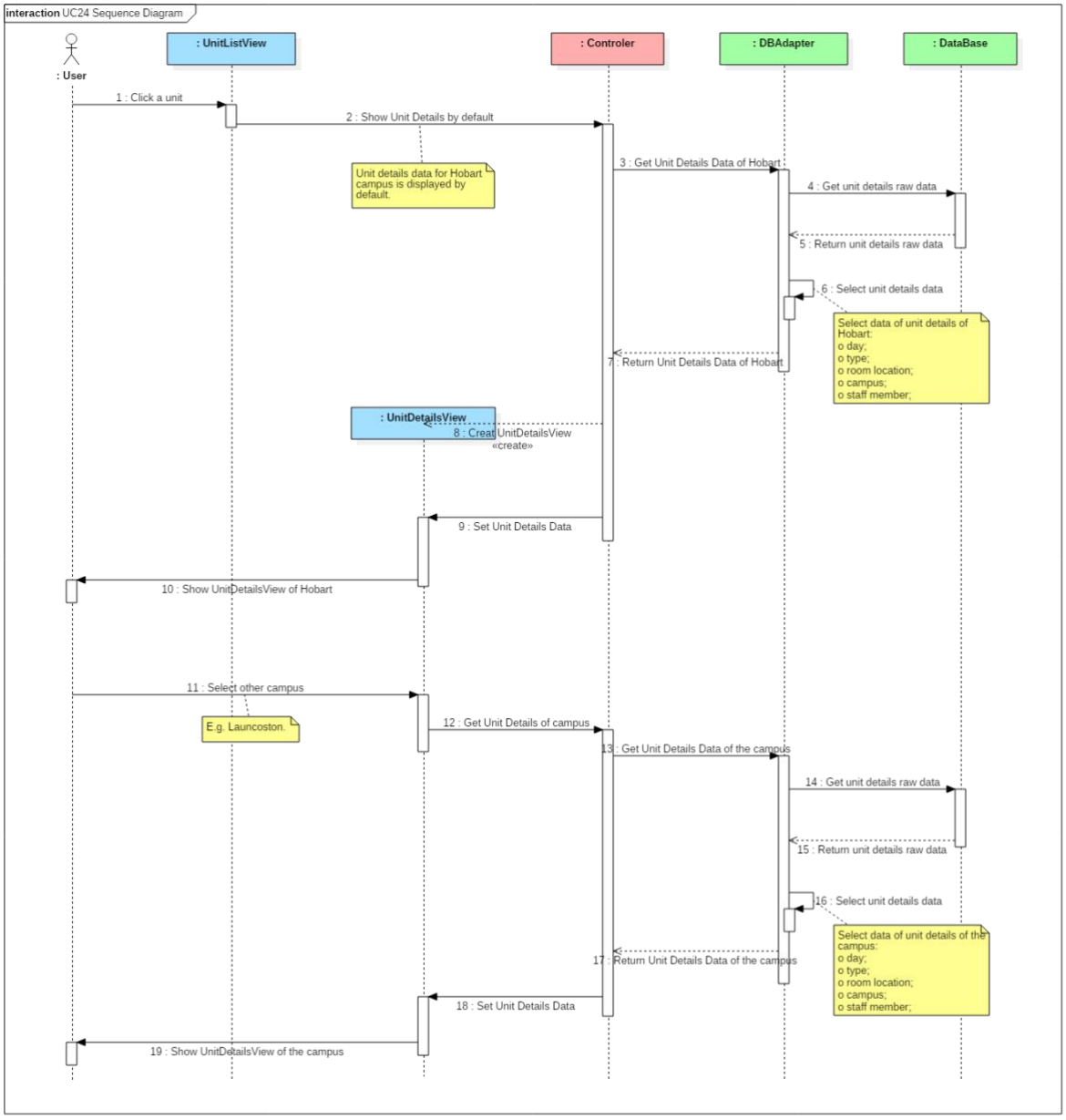
5.3. Sequence diagram of UC19_User_shows_ActivityGrid



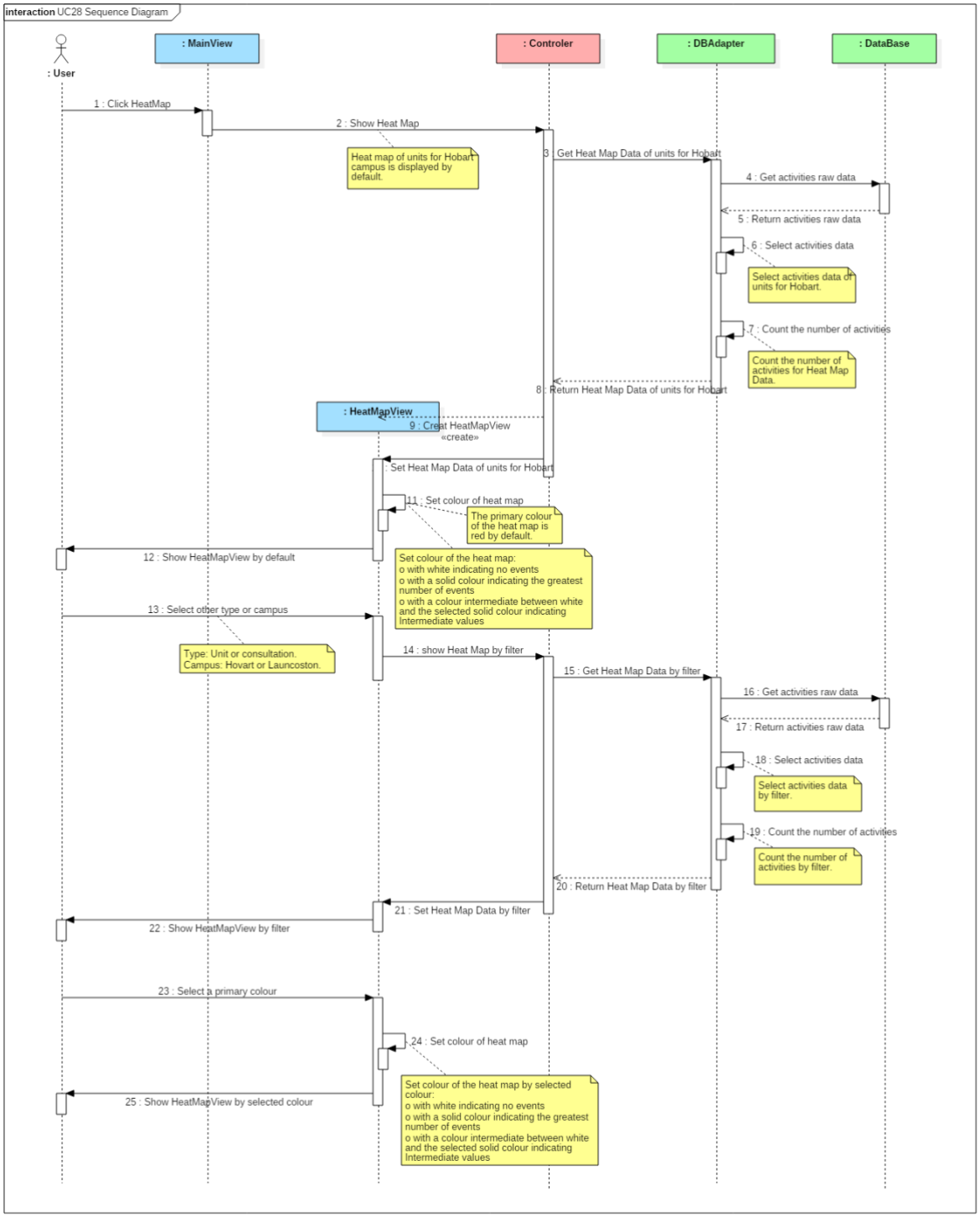
5.4. Sequence diagram of UC23_User_views_UnitList



5.5. Sequence diagram of UC24_User_selects_Unit



5.6. Sequence diagram of UC28_User_generates_HeatMap



5.7. Sequence diagram of UC37_User_generates_ClashMap