HEURISTIC EVALUATION OF DESIGN PROJECT-: IMPROVING EFFICIENCY OF CENTRE FOR STUDENTS WITH DISABILITIES

TEAM MEMBERS:

- Fasuyi, Morounkeji
- Leong, Madeleine Min Jing
- Neveditsin, Nikita
- Sadi, Sadman Hoque
- Shree, Bhagya
- Tong, Xinyun

TABLE OF CONTENTS

TEAM EVALUATION

Heuristics Evaluation of [Design of the project: Improving Efficiency of Centre for Students with Disabilities**]**

By Team **Date** November 10, 2018

Brief explanation of a scenario (filled by the group who designed UI)

To View student Information/Course/Schedule

1. Login to Web application
    - 1.1 Launch Web application
    - 1.2 Typing username in username text box
    - 1.3 Typing password in password text box
    - 1.4 Click “Login” button
2. Navigate to “Student List” Form
    - 2.1 Click “Student List” tab
3. Locate the student entry
    - 3.1 Find student in “Student List” Form
    - 3.2 If the student you looking for is not in the current page, click the page number on the left bottom corner or click the next page button
    - 3.3 Click the “View” link after the student name

      OR
    - 3.1 Type student’s name in the search box located at the right upper corner
    - 3.2 Press “Enter” on your keyboard
    - 3.3 The match record will be highlight on the "Student List” form
4. View the Personal Information and the Students course enrolled
    - 4.1 Student information will display in the “Student Info” Form
    - 4.2 Click drop down list after “Courses Enrolled” field to view the enrolled courses
5. Check the student schedule in Calendar
    - 5.1 Click “Check Schedule” button on the right bottom corner
    - 5.2 The pop-out calendar will show the detail schedule
6. Go back to the homepage.
    - 6.1 Click “Back” button on the left bottom corner to go back to the homepage
7. Log out from the application
    - 7.1 Click “Logout” link on the right upper corner to log out from the application

1. Visibility of system status

- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

Evaluation

General Observations

1. Student List tab clearly lists all the students enrolled along with button for further viewing detailed information.
2. There are currently no status display features added to the design.
3. We also have search bar at the top right corner, to just type the students name to search.
4. Student Information is visible clearly except for course enrolled field

Suggested Solutions:

1. The semester info appears on the homepage but not at the student list.
2. 'Sign Up' option is not displayed prominently; the text size is smaller and makes UI unintuitive for first time users who are most likely to use this page
3. Search option in 'Student List' page should be in the same cluster as the list.
4. In homepage add dashboard module/tab to shows how many schedules for today, how many new schedules received from student etc.
5. Back button in the home screen is unnecessary since this is the starting page and browser already provides a back button.
6. Forgot password option needs to be introduced.
7. The course enrolled field should not be a dropdown box rather, it should be in tabular form as to which courses the students are enrolled.
8. Log out is just present at the home page, it should also be visible in all screens.
9. Tabs student list. Schedules, alerts, test/exam should look more like buttons
10. In "Student List" form search box shows message "No Record Found" if there is no match
11. A loading icon when a user logs in will be added to show the process is being carried out.
12. Every click action will be accompanied by a loading icon to show that the process is taking place because depending on internet speed, load time may vary for users and they have no way of knowing whether to wait or click repeatedly.
13. Username who logged in should be reflected in all pages

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

General Observations:

1. The language used in the design are real world terms like alerts, check schedules, student list. The target user should be familiar with all the terms used
2. The Design of the field is very simplistic and easy to understand by user.
3. Location of each tab and the description of each field is easy to understand.

Suggested Solutions:

1. In “Student Info” form change “PhoneNo” to “Phone”
2. Set alert for student or staff
3. Alert field should be renamed to alert frequency
4. There should be a setting that specifies events that may cause alerts
5. Rename Test/Exam to Test/Exam schedules

3. User control and freedom

- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

Evaluation

General Observations:

1. The tabs on the Home page are closely located and hence giving space between tabs will give user freedom to easily move around tabs.
2. Student name can be embedded with link because, the user can get confused with whether or not to click view or name to go to the next screen.
3. Search button is too small

Suggested Solutions:

1. A back/cancel button should be added to all the pages/popup windows
2. When the user presses an exit or cancel button, there should be a pop-up dialog box to confirm action. This would provide the freedom to undo any accidental actions.
3. There should be a static home button on every page to go back to the home screen at any moment
4. Student list should have add/edit/delete buttons

4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

1. On all the pages apart from the home page, there is no consistent way to return to the home page.
2. The buttons that mean the same thing have the same wordings, color and locations across all the pages so it is easy for a user to follow.
3. From the homepage, user may think VIEW and SCHEDULE is the same thing because he/she may think both are to view schedule for test classroom and availability.
4. A small cross button at the homepage and the Logout word at the right top of the home page contain the same meaning for user to logout
5. 'View' is typically a part of 'settings' but are represented as separate on home screen

5. Error prevention

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Evaluation

1. At the login page, an error message should be shown if username and password are mismatch or no existence of the account.
2. On delete action for either schedules or student, confirmation should be required.
3. Home screen has a lot of buttons which can easily cause confusion.
4. Field Validations for all forms

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

General Observations:

1. No clear indication of where in the website the user is causing it difficult to understand how to go back to a specific page.
2. Number of buttons on home screen makes it hard to remember which option is where.

Suggested Solutions:

1. The options at the vertical tab should be changed to some other words to let user more understandable at the first glance
2. Tab flow (E.g. Home - > Student Track -> Student Schedule) should be reflected on each screen
3. Use scrollbar instead page button in "Student List" module
4. List all enrolled courses in "Student Info" form rather than drop list
5. Combine calendar view and list view when click "Check Schedule" button in "Student Info" form;
6. Mouse move to the pop out calendar window in "Student Info" form to shows a drop-down list of schedules, click schedule enter schedule detail form
7. Clear headings should be available on every page

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

General Observations:

1. Lack of obvious flow of content makes it difficult for users to hit 'tab' to go through all the buttons and boxes.
2. The Portal seems flexible enough, for both new or the frequent user
3. The portal gives the search bar for efficiently search the students if the list of students is too much, which will be painful to scan through pages

Suggested Solutions:

1. Add a filter and sort feature on the top of "Student List" module, it can filter Name First, Name Last, A#, Phone, Email etc.
2. Add a menu item 'Recent' with the topmost 5-10 recent things that the user has performed.

8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

General Observation:

1. On the homepage, there are two buttons: back arrow and cross. This may lead to confusion for the user. The visibility of Logout is not clear.
2. Info about correct date and time and current term is not necessary
3. All major headings are easy to understand. It is also easy to scan through the UI. Clear terminology is implemented all across the program.

Suggested Solutions:

1. Add "Add Student", "Delete Student" features to the left hand menu under "Student List" module/tab;
2. Move "Alerts" menu under "Setting" tab
3. - Add "Add Schedule", "Delete Schedule", "Modify Schedule" features to the left-hand menu under "Schedule" Module/tab

9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

General Observation:

No error handling features were included in the design

Suggested Solutions:

1. App should have a log file to help support staff
2. The error messages will also be specific about the exact reason for the error
3. In "Student Info" form if there is an error found during validation process, highlight error field in red color and shows requirement after the field

10. Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

General Observations:

1. Although we hope that a user does not have to user this feature because the interface should be easy to navigate through, we provided a help button which is on the Menu bar of the home page.
2. Users can find documentation, or contact support through this icon.
3. A help button is available at the homepage only.
4. A small introduction about the portals functionality on home page will be good
5. - There are no assist or explanation pop-ups for individual options.

TEAM MEMBER – FASUYI MOROUNKEJI

By Fasuyi Morounkeji - A00421483 **Date** 11/9/18

1. Visibility of system status

- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

Evaluation

Observations:

There is a heading display of the current page the user is on.

There are currently no status display features added to the design.

Solutions:

Logging on to the system: A loading icon when a user logs in will be added to show the process is being carried out.

Every click action will be accompanied by a loading icon to show that the process is taking place because depending on internet speed, load time may vary for users and they have no way of knowing whether to wait or click repeatedly.

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

Observation:

The language used in the design are real world terms like alerts, check schedules, student list. The target user should be familiar with all the terms used.

3. User control and freedom

- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

Evaluation

Observation:

Student List Display: This form has no undo/back button. This leaves the user with no way to leave an unwanted state unless to press the browsers back button functionality.

Solution:

A back/cancel button will be added to all the pages, with the same colors and in the same locations so that it is easily identifiable by the user.

Also, when the user presses an exit or cancel button, there will be a pop-up dialog box to confirm action. This would provide the freedom to undo any accidental actions.

4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

Observation:

On all the pages apart from the home page, there is no consistent way to return to the home page.

The buttons that mean the same thing have the same wordings, color and locations across all the pages so it is easy for a user to follow.

Solution:

On all the pages, there will be a consistent way to return to the home page. Also, a consistent back button or cancel button will be added to all pages.

5. Error prevention

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Evaluation

Solution:

All the fields for data entry would be validated to prevent errors. For example, entering the A# of students, a text will be printed out to the right of the field showing the text validations like if it starts with an A and is 8 digits.

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

Observation:

All the buttons needed are provided at locations that are easily identifiable or that follows the standard. For example, the menu bar.

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

Observation:

On checking the student schedule, when a user hovers over a particular date, the user can easily see the class schedule for the student. It is an easy and fast way of knowing the student schedule without actually going to view the total student class schedule.

Also, the student list provides a sort icon that is easily identifiable by the user to sort based on the user's preference or needs at the moment. Maybe by most recent or names in alphabetical order.

8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

Observation:

The design of the pages only includes information that is pertinent to the purpose of the user on the page. It is clutter free and every information displayed on the interface is necessary and useful.

All major headings are easy to understand. It is also easy to scan through the UI. Clear terminology is implemented all across the program.

9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

Observation:

No error message features were included in the design

Solution:

Easy to understand real world text would be printed out as error messages for the user to understand why a particular action didn’t happen as it should have.

The error messages will also be specific about the exact reason for the error.

10. Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

Observation:

Although we hope that a user does not have to user this feature because the interface should be easy to navigate through, we provided a help button which is on the Menu bar of the home page.

Users can find documentation, or contact support through this icon.

It is also easily accessible as it is displayed on the home page menu where it is usually located for other applications that the user would have previously been familiar with e.g. Microsoft Word.

TEAM MEMBER – SADMAN HOQUE SADI

By: Sadman Hoque Sadi

1. Visibility of system status

- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

Evaluation

[Enter your observation and evaluation of the degree to which this Heuristic has been satisfied. Use as much space as you see fit.]

1. The 'Sign in' and 'Sign Up' options are in the same cluster.
2. 'Sign Up' option is not displayed prominently; the text size is smaller and makes UI unintuitive for first time users who are most likely to use this page.
3. Back button in the home screen is unnecessary since this is the starting page and browser already provides a back button.
4. The top bar consisting of 'home' and 'logout' is not static.
5. 'View' button in student list is unintuitive as clicking on the name itself comes more naturally.
6. Search option in 'Student List' page should be in the same cluster as the list.

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

1. There is nothing wrong with the language usage.

3. User control and freedom

- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

Evaluation

1. There should be a static home button on every page to go back to the home screen at any moment.
2. No option to go back 'up' to the page right before current one.

4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

1. 'view' is typically a part of 'settings' but are represented as separate on home screen

5. Error prevention

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Evaluation

1. Home screen has a lot of buttons which can easily cause confusion.

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

3. No clear indication of where in the website the user is causing it difficult to understand how to go back to a specific page.
4. Number of buttons on home screen makes it hard to remember which option is where.

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

1. Lack of obvious flow of content makes it difficult for users to hit ‘tab’ to go through all the buttons and boxes.
2. Drop down menus are inherently slow.

8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

1. Home screen has a lot of unimportant functionality displayed prominently.

9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

10. Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

1. Help is only available on the home screen.
2. There are no assist or explanation pop-ups for individual options.

UI/UX Design and Evaluation Heuristic Evaluation

TEAM MEMBER – TONG XINYUN

By Tom Tong **Date** 9 Nov 2018

1. Visibility of system status

- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

Evaluation

[Enter your observation and evaluation of the degree to which this Heuristic has been satisfied. Use as much space as you see fit.]

1. In homepage add dashboard module/tab to shows how many schedules for today, how many new schedules received from student etc.;
2. In "Student List" form search box shows message "No Record Found" if there is no match record;

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

1. In "Student Info" form change "PhoneNo" to "Phone";
2. In "Student Info" form change "Emergent Contact" to "Emergency Contact";

3. User control and freedom

- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

Evaluation

1. In "Student Info" form, add "Cancel" button to exit without save changes;

4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

N/A

5. Error prevention

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Evaluation

1. In "Student Info" form validate A# should begin with A and follow by only 8 digit number ;
2. In "Student Info" form validate PhoneNo should only contain number;
3. In "Student Info" form validate email should follow email format standard;
4. In "Student Info" form validate Emergency Contact should only contain number;

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

1. Use scollbar instead page button in "Student List" module;
2. List all enrolled courses in "Student Info" form rather than drop list;
3. Combine calendard view and list view when click "Check Schedule" button in "Student Info" form;
4. Mouse move to the pop out calendard window in "Student Info" form to shows a drop down list of schedule, click schedule enter schedule detail form;

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

1. Add a filter feature on the top of "Student List" module, it can filter Name First, Name Last, A#, Phone, Email etc and also able to click to sort by field;
2. Double click student to open "Student Info" form rather than click view link;

8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

1. Put "Student List" form into homepage working area;
2. Move "Student List" Menu in the homepage to the top tab, so the tab (on the top) is module and the menu (on the left) is function;
3. Merge "Test/Exam" into "Schedule" so "Schedule" included "Course" and "Exam";
4. Move "Alerts" menu under "Setting" tab;
5. Add "Add Student", "Delete Student" features to the left hand menu under "Student List" module/tab;
6. Add "Add Schedule", "Delete Schedule", "Modify Schedule" features to the left hand menu under "Schedule" Module/tab;

9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

1. In "Student Info" form if there is an error found during validation process, highlight error field in red color and shows requiement after the field;

10. Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

N/A

UI/UX Design and Evaluation Heuristic Evaluation

TEAM MEMBER – MADELEINE MIN JING LEONG

By Madeleine Min Jing Leong **Date** November 8, 2018

1. Visibility of system status

- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

Evaluation

[Enter your observation and evaluation of the degree to which this Heuristic has been satisfied. Use as much space as you see fit.]

At the Student List tab, the semester information is appeared on the homepage but not at the student list.

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

The horizontal and vertical tabs shown at the homepage may lead to confusion to first-time users because they may not exactly know what the function of horizontal tab is if all features they want are at the vertical tab.

3. User control and freedom

- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

Evaluation

When user goes to the student list page, there is no button for user to go back to the homepage

If user want to search the student, did he/she search by name? or search name on the shown page only?

4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

From the homepage, user may think VIEW and SCHEDULE is the same thing because he/she may think both are to view schedule for test classroom and availability.

A small cross button at the homepage and the Logout word at the right top of the home page contain the same meaning for user to logout?

5. Error prevention

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Evaluation

At the login page, an error message should be shown if username and password are mismatch or no existence of the account.

An error of unfound student on the search should be considered.

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

The options at the vertical tab should be changed to some other words to let user more understandable at the first glance.

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

A homepage button is not available for user to go back to the homepage. This can be simply solved by remaining the vertical tabs to every page.

8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

On the homepage, there are two buttons: back arrow and cross. This may lead to confusion for the user. The visibility of Logout is not clear.

9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

When user insert wrong student name on the student list, there is no back or research button available.

10. Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

A help button is available at the homepage only.

TEAM MEMBER – BHAGYA SHREE

By (Bhagya Shree) **Date** 8th November 2018

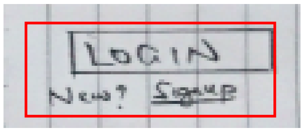
1. Visibility of system status

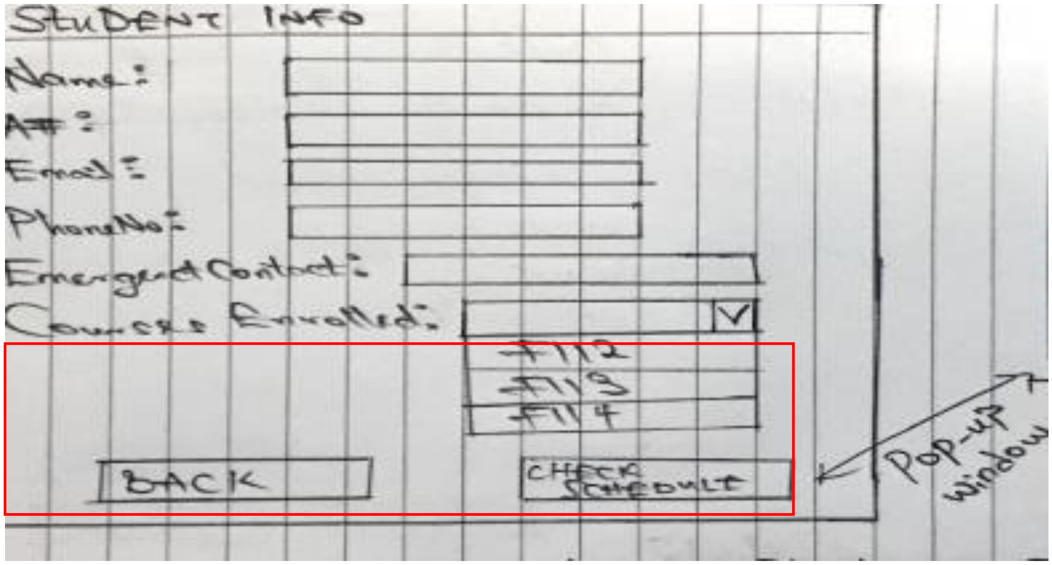
- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

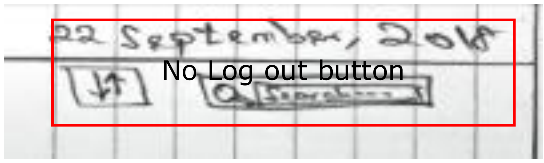
Evaluation

[Enter your observation and evaluation of the degree to which this Heuristic has been satisfied. Use as much space as you see fit.]

1. Login page is clear and straightforward along with sign up.
2. Forgot password option needs to be introduced.
3. Home Page, we can easily locate Student List.
4. Student List tab clearly lists all the students enrolled along with button for further viewing detailed information.
5. We also have search bar at the top right corner, to just type the students name to search.
6. Student Information is visible clearly except for course enrolled field
7. The course enrolled field should not be a dropdown box rather, it should be in tabular form as to which courses the students are enrolled.
8. Check Schedule displaying a pop-up calendar and highlighting students schedule is good.
9. Go back button is clear, it can be located at the upper right corner.
10. Log out is just present at the home page , it should also be visible in all screens.







No Log out button

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

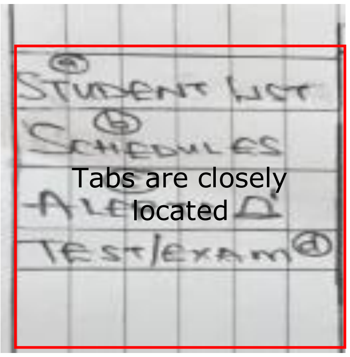
1. Naming Convention of the fields is understandable by the common man.
2. The Design of the field is very simplistic and easy to understand by user.
3. Location of each tab and the description of each field is easy to understand.

3. User control and freedom

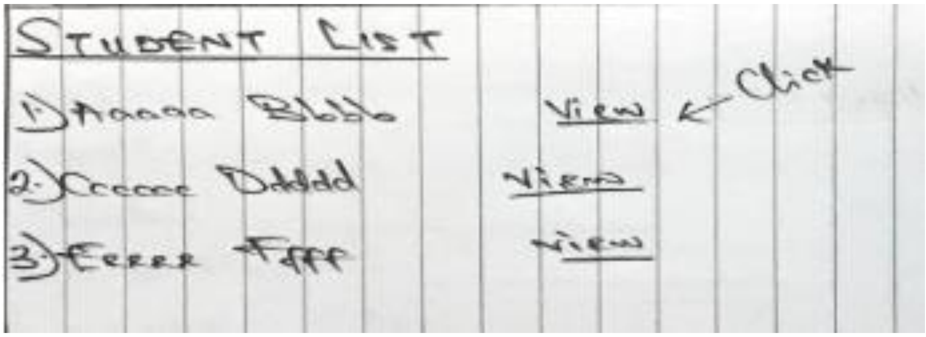
- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

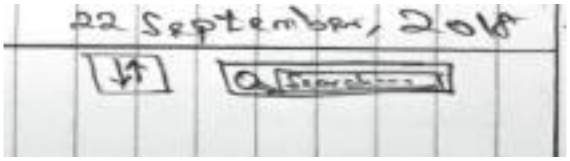
Evaluation

1. There is no option for user to login, if forgot the password.
2. The tabs on the Home page are closely located and hence giving space between tabs will give user freedom to easily move around tabs.
3. Student name can be embedded with link because, the user can get confused with to click view or name to go next screen.
4. Search button is too small.



Tabs are closely located





4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

1. The header bar (Home, View, Settings, Tab) is not present in all pages.
2. Back button is not present in all pages.
3. Username who logged in should be reflected in all pages.

5. Error prevention

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Evaluation

1. Going back to home page is not possible directly from any page.
2. Back button is not present in all pages.
3. The option to add, delete the students from student list is not present.

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

1. Tab flow (For ex : Home - > Student Track -> Student Schedule)should be reflected on each screen
2. Back button is not present in all pages.

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

1. The Portal seems flexible enough, for both new or the frequent user
2. The portal gives the search bar for efficiently search the students if the list of students is too much, which will be painful to scan through pages

8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

1.Pages have minimalistic design, which are used by the user most of the time.

9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

1. The option of editing student information is not present
2. The option to add ,delete the students from student list is not present

10. Help and documentation

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

1. Help documentation is not present.
2. A small introduction about the portals functionality on home page will be good

TEAM MEMBER – NIKITA NEVEDITSIN

By (your name) Nikita Neveditsin **Date** 2018-11-09

1. Visibility of system status

- Always keep users informed about what is going on.
- Provide appropriate feedback within reasonable time.

Evaluation

[Enter your observation and evaluation of the degree to which this Heuristic has been satisfied. Use as much space as you see fit.]

⊖ Tabs 'student list', 'schedules', "alerts', 'test/exam' should look more like buttons (that we can click on them)

⊖ after clicking on one of the buttons/links the system should show that they are clicked (change font size or show temporary "loading" pages if loading from database takes long time)

⊖ not defined how to select a term. Do we need to select a term?

2. Match between system and the real world

- Speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
- Follow real-world conventions, making information appear in a natural and logical order.

Evaluation

<2c> - set alerts.
For who? For students or for staff?
- (Alert) field should be renamed to "Alert frequency"?
- There should be a setting that specifies events that may cause alerts.

Overall, there is no technical words/phrases that could not be understanded by users.

- <2d> - should rename 'Test/Exam' to "Test/Exam schedules"

3. User control and freedom

- Users often choose system functions by mistake.
- Provide a clearly marked "out" to leave an unwanted state without having to go through an extended dialogue.
- Support undo and redo.

Evaluation

- Each tab should have a link to homepage
- Student list should have 'add'/ 'delete'/ 'edit' options (2 a)
- Each pop-up window should have 'cancel' button (2c - alerts)
- Support 'system backup' to ~~A~~ hr in the past? For 1week?

4. Consistency and standards

- Users should not have to wonder whether different words, situations, or actions mean the same thing.
- Follow platform conventions.

Evaluation

- What do icons 14 , 88 mean? (Page ② - Home page)
Maybe move them to other tab?
Do as buttons with ~~te~~ text?

5. Error prevention

- Even better than good error messages is a carefull design which prevents a problem from occurring in the first place.

Evaluation

- When somebody tries to delete /edit student info, ask them 'do they really want to delete student?'
- When we delete scheduled events (exams), ask them for confirmation

6. Recognition rather than recall

- Make objects, actions, and options visible.
- User should not have to remember information from one part of the dialogue to another.
- Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Evaluation

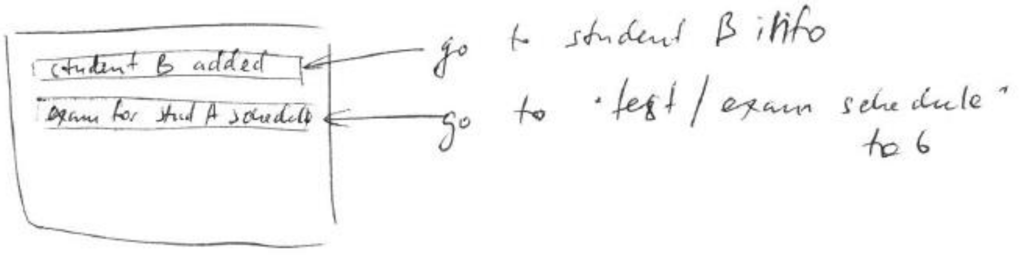
- 26 - on Schedules tab. Do we need it? We can access each student schedule from Student Info tab, so "Schedules" (26) is probably redundent

7. Flexibility and efficiency of use

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users.
- Allow users to tailor frequent actions.

Evaluation

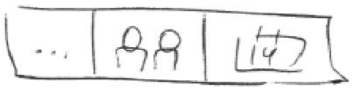
- Maybe we can add a menu item "Recent" with links to 5-10 most recent tabs accessed:

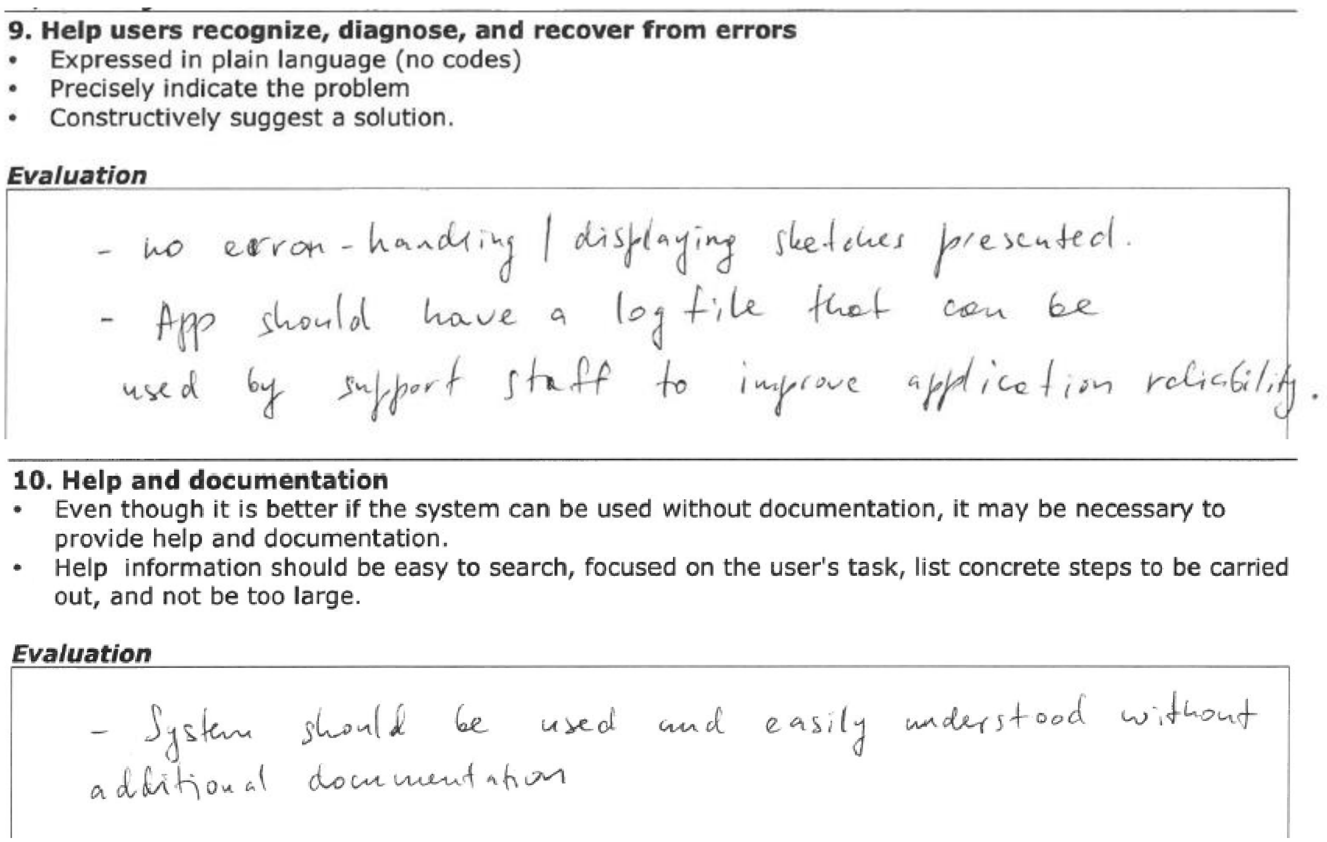


8. Aesthetic and minimalist design

- Dialogues should not contain information which is irrelevant or rarely needed.
- Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Evaluation

- view is redundant.
-  is hard to understand / non-descriptive
- information about current date/time and current term is not necessary.
- Student list - maybe "students" should be clickable?



9. Help users recognize, diagnose, and recover from errors

- Expressed in plain language (no codes)
- Precisely indicate the problem
- Constructively suggest a solution.

Evaluation

- no error-handling / displaying sketches presented.
- App should have a log file that can be used by support staff to improve application reliability.

10. Help and documentation

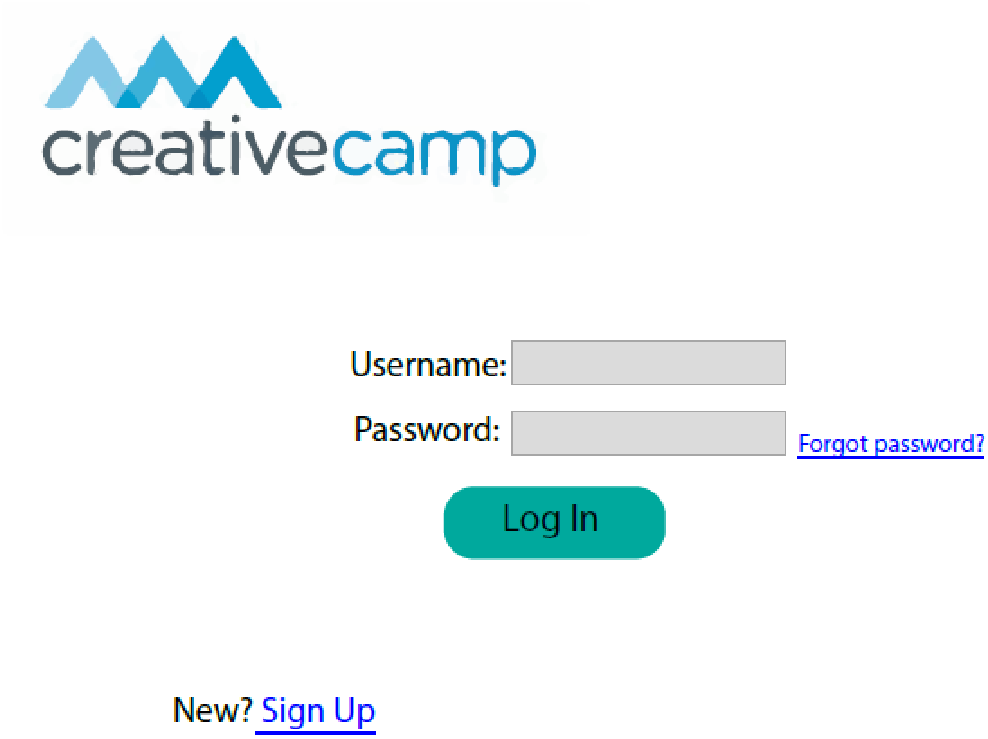
- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.
- Help information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Evaluation

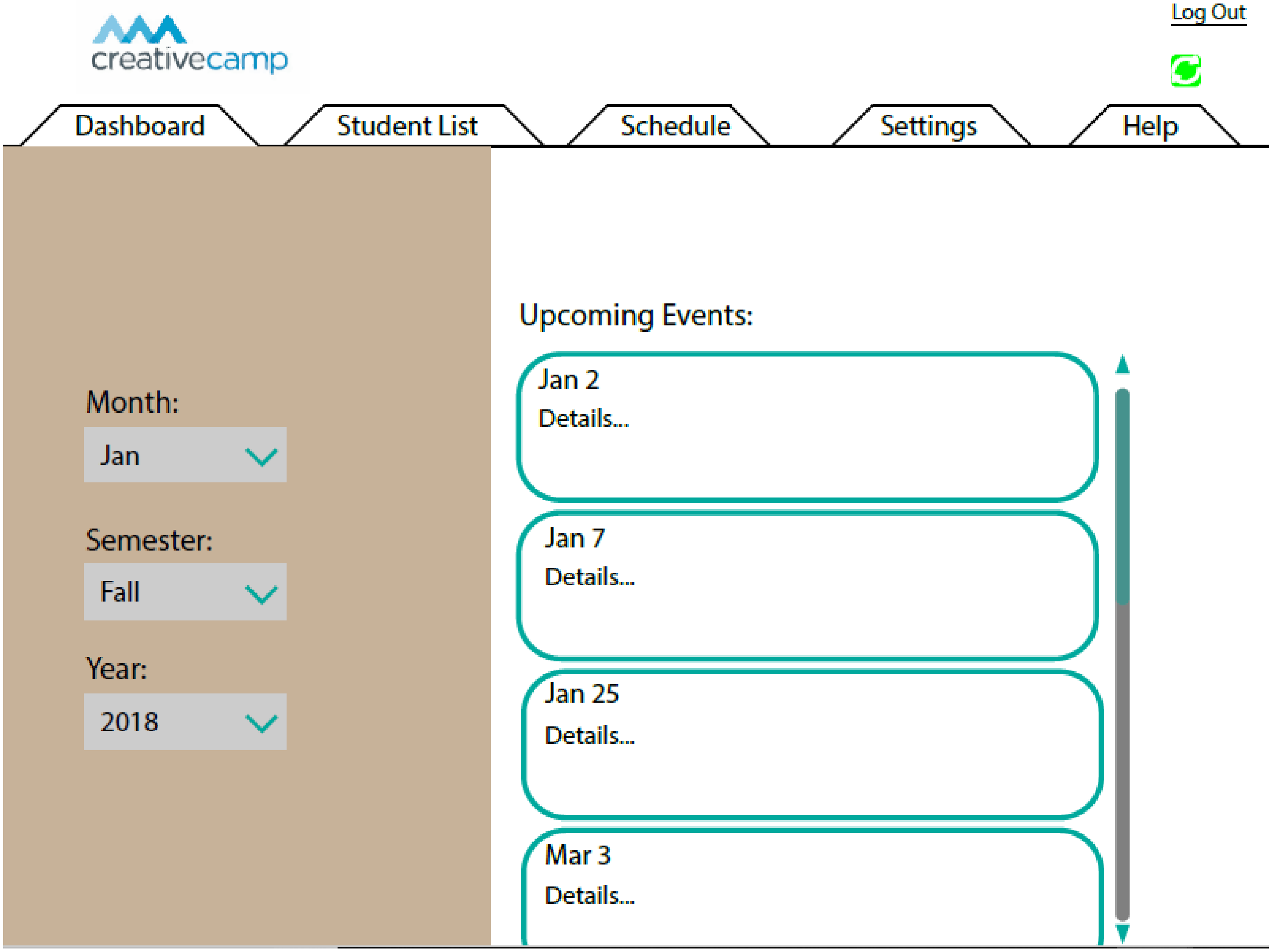
- System should be used and easily understood without additional documentation

REVISED SKETCHES

1. ***Login Page*** - The first page a user sees on launching the application.



2. ***Dashboard*** – This is the landing page after a user logs on to the system



Month: Jan

Semester: Fall

Year: 2018

Upcoming Events:

Jan 2
Details...

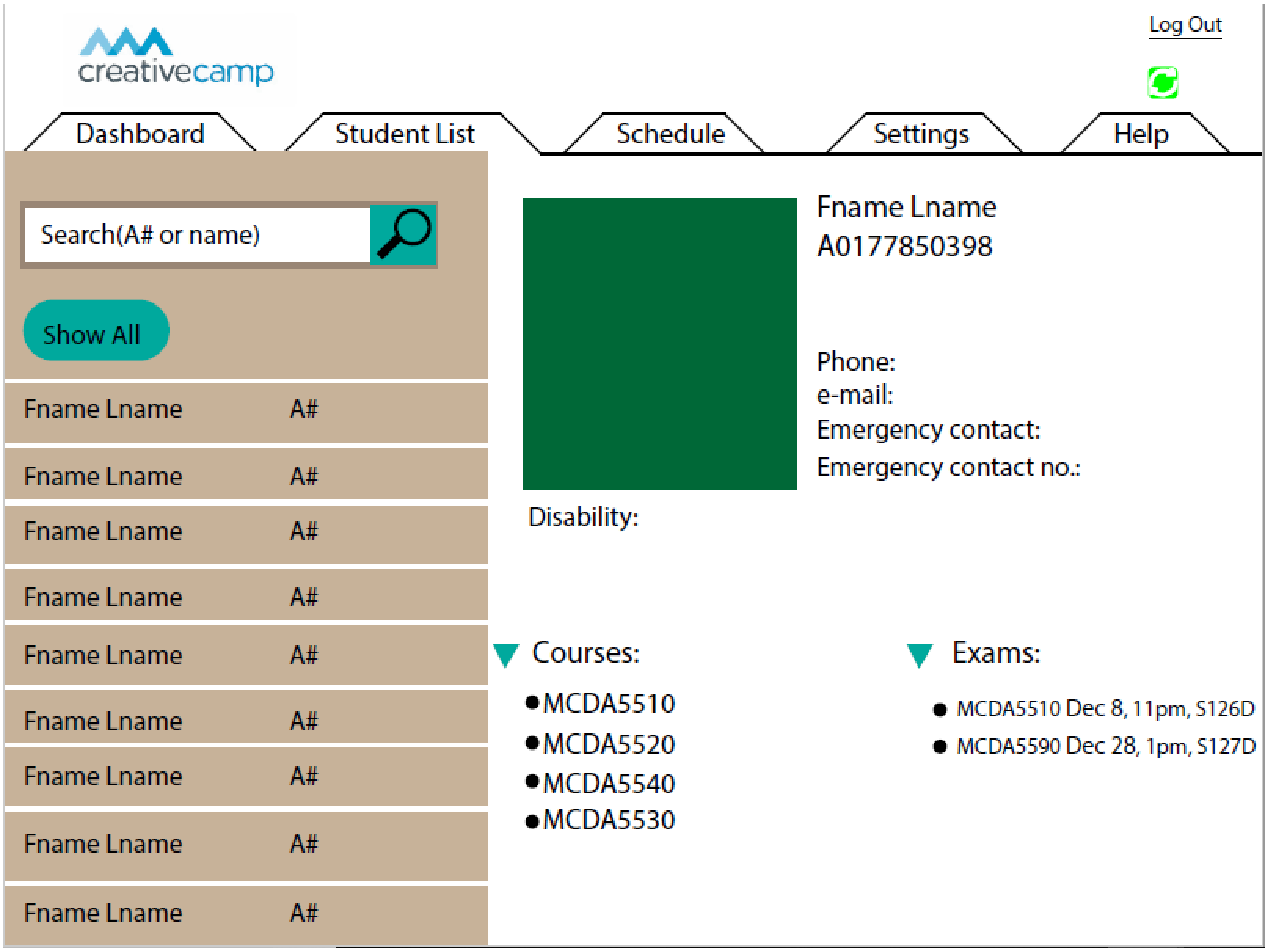
Jan 7
Details...

Jan 25
Details...

Mar 3
Details...

3. ***Student List*** – On this page, a user can search for student by either name or A#. Drop down selections will be used to make the process faster. As the user types, entries that match will populate the drop-down list till the user finds the intended student.

   Once the student is selected the grid on the right is populated with the student details.



Fname Lname	A#
Fname Lname	A#
Fname Lname	A#
Fname Lname	A#
Fname Lname	A#
Fname Lname	A#
Fname Lname	A#
Fname Lname	A#
Fname Lname	A#

Fname Lname
A0177850398

Phone:
e-mail:
Emergency contact:
Emergency contact no.:

Disability:

▼ Courses:

- MCDA5510
- MCDA5520
- MCDA5540
- MCDA5530

▼ Exams:

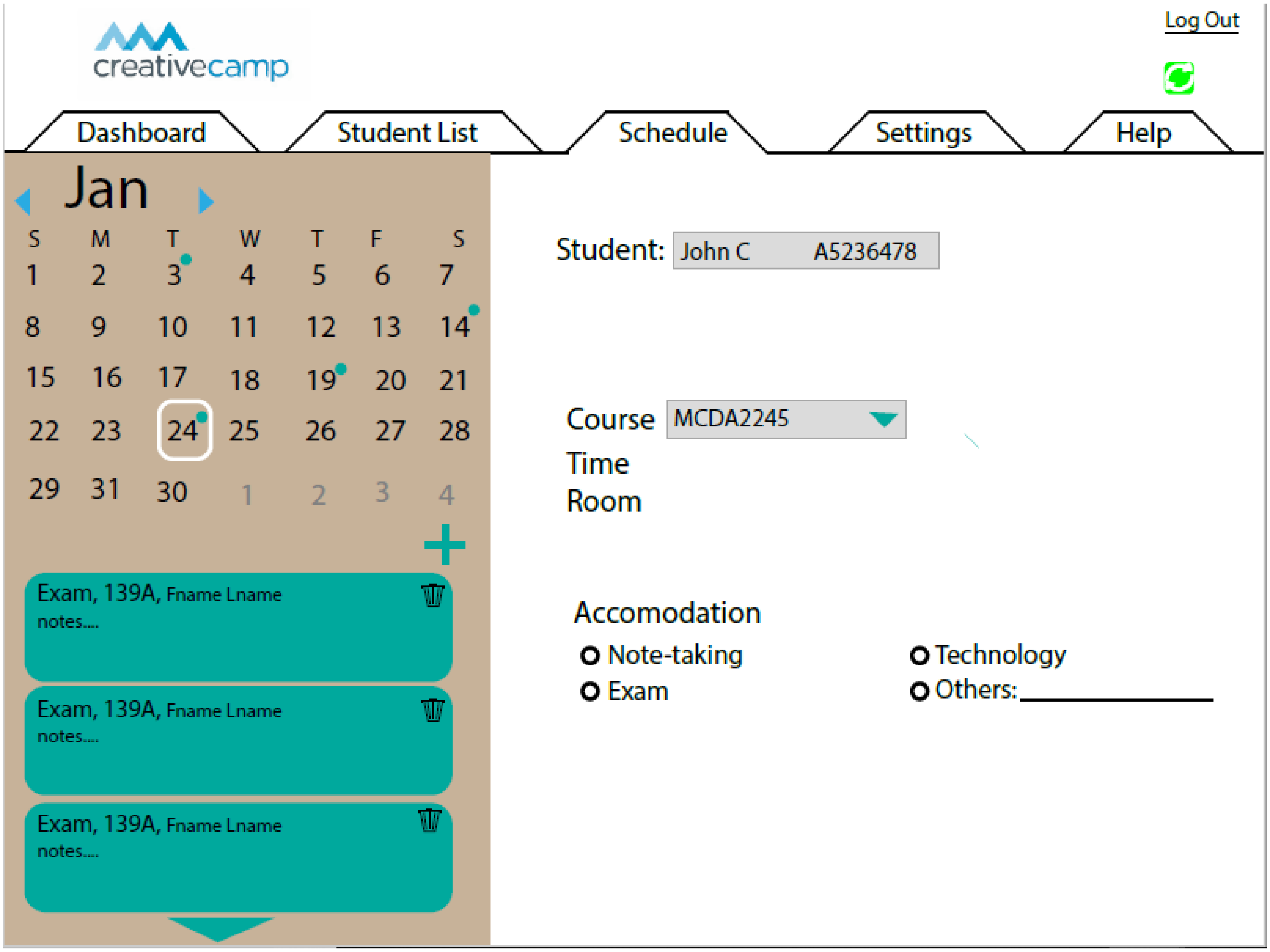
- MCDA5510 Dec 8, 11pm, S126D
- MCDA5590 Dec 28, 1pm, S127D

4. ***Schedule*** – On this page, the user can view scheduling information for test and exams or tasks for the day.

   The user can also add a new schedule. Once the user gets a request for accommodations from a student, the user can input the date and time of the test or exam, include a room available for the student and also keep track of any other accommodations the student might need so it can be provided as at when due like note takers, extra technology etc.

   Similar to the student list search bar, as the user enters a name or A#, the system searches he database and populates a drop down with matching names till the intended target is found.

   All the schedules are stored in the calendar and days with events are denoted with the dots much like the phone calendars.



S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	31	30	1	2	3	4

Exam, 139A, Fname Lname
notes....

Exam, 139A, Fname Lname
notes....

Exam, 139A, Fname Lname
notes....

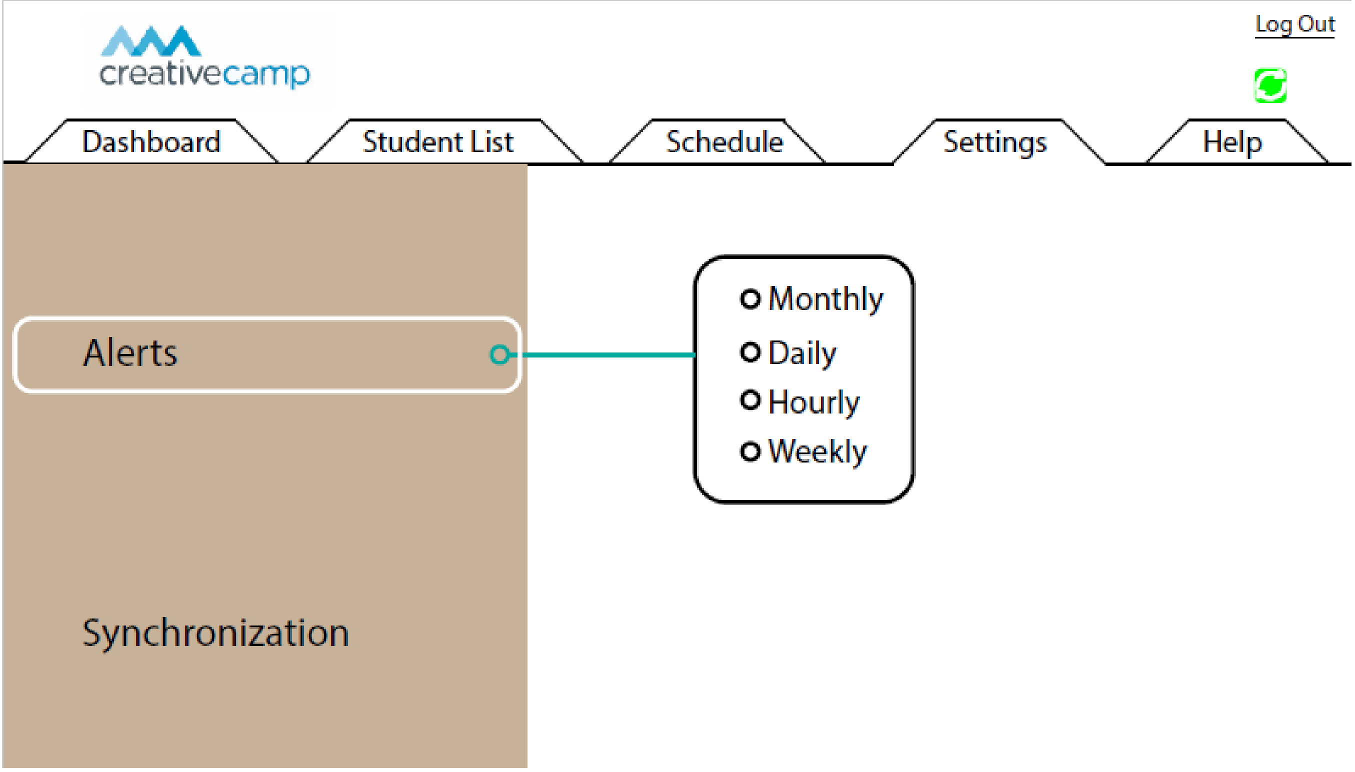
Student: John C A5236478

Course MCDA2245
Time
Room

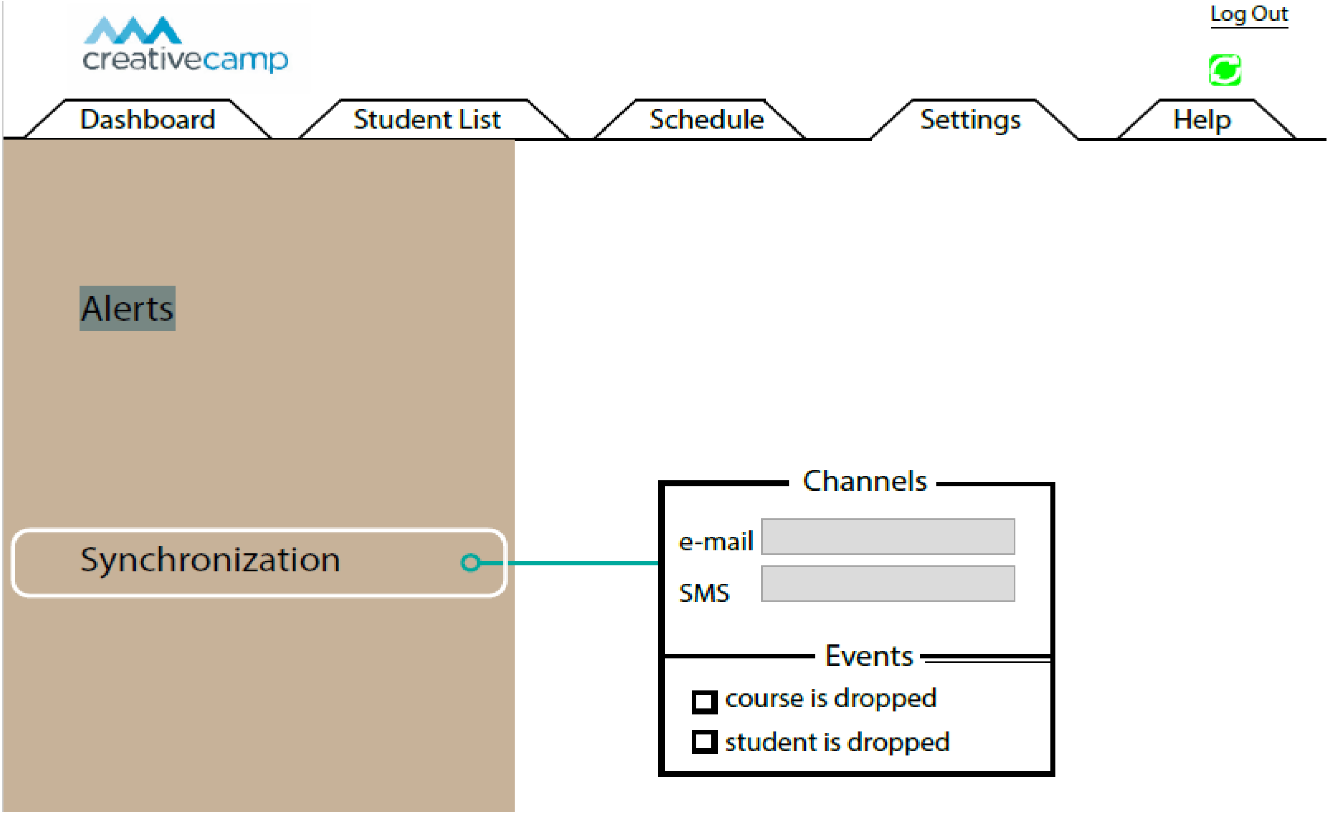
Accomodation

○ Note-taking ○ Technology
○ Exam ○ Others:________________

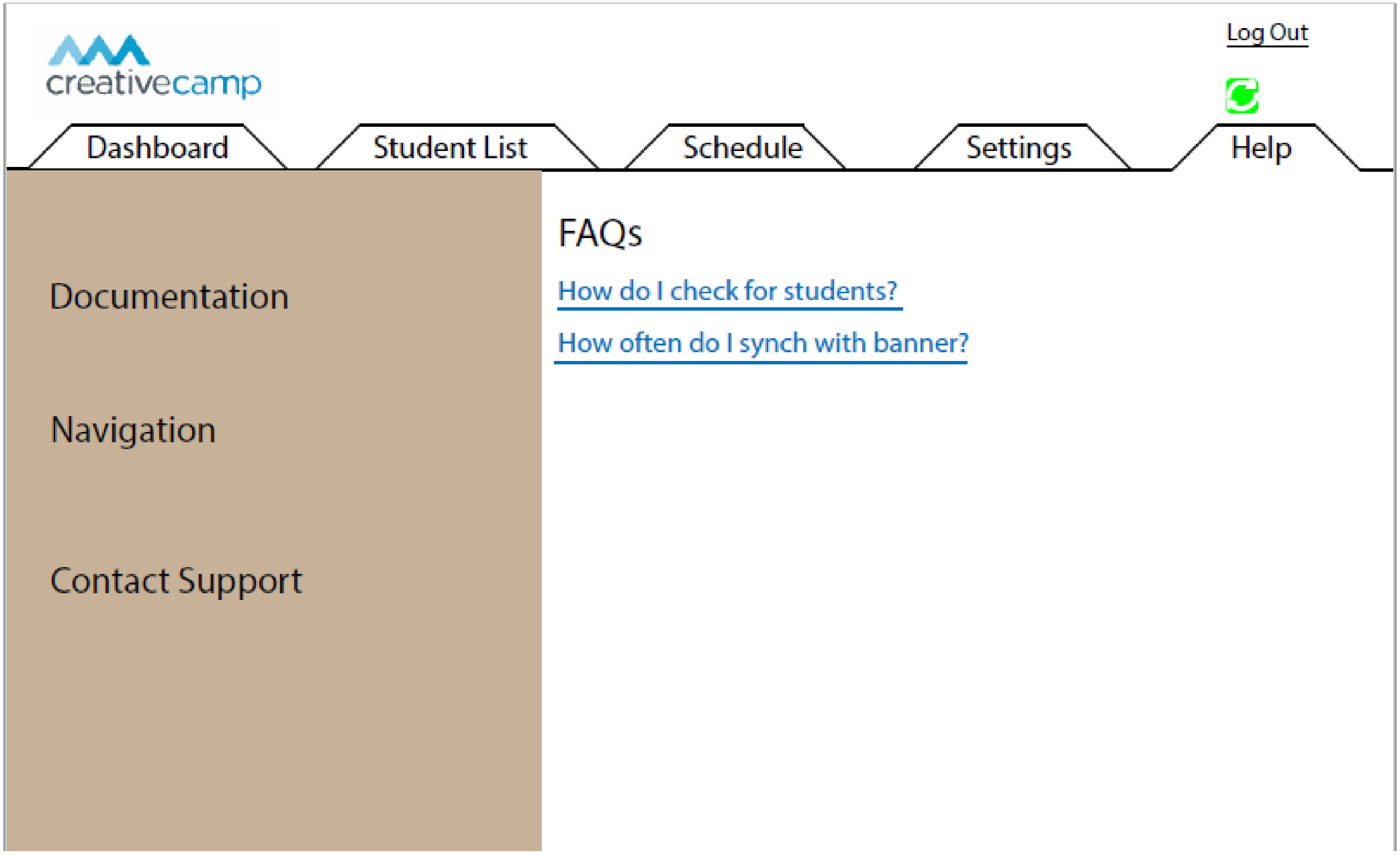
5. ***Settings –*** This page is for users to configure some functionality of the system.
   The frequency of the alerts received by the user either by email or SMS can be configured.



Also, the synchronization with the banner to get new information about any changes can be configured automatically



6. ***Help -*** Although we hope that a user does not have to use this feature because the interface should be easy to navigate through, we provided a help page for user support.

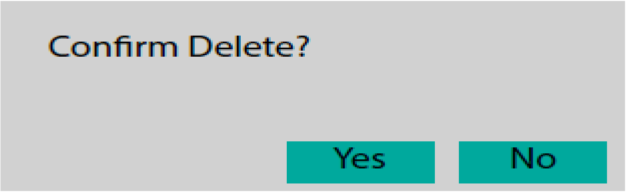


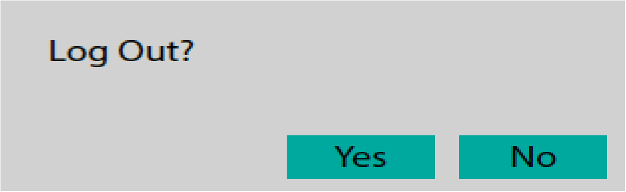
FAQs

How do I check for students?

How often do I synch with banner?

7. ***Error Prevention –*** This is a confirmation dialog box that will pop-up after any delete action to prevent any accidents.


Confirm Delete?


Log Out?