

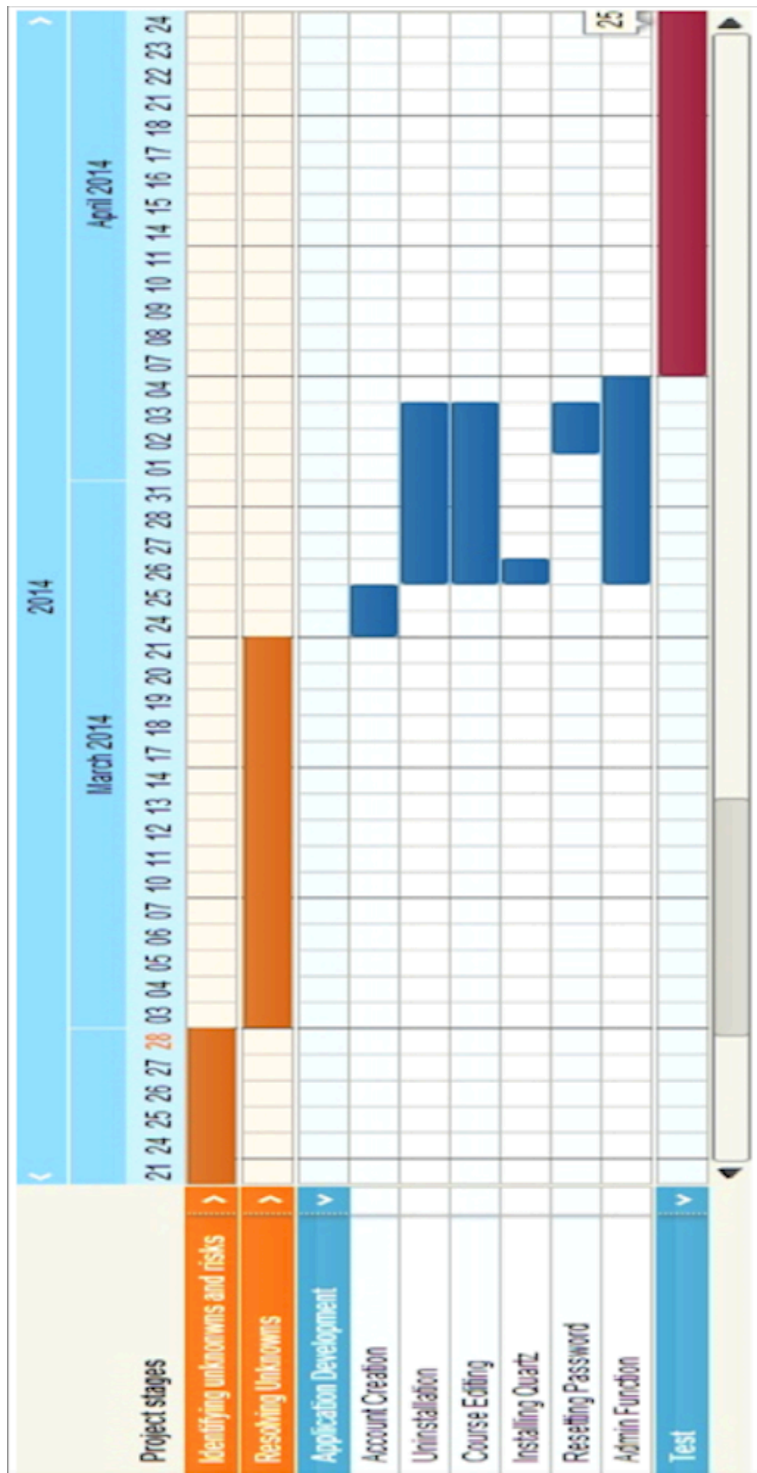
Cynthia Chan

CS411

Group Status Report:

The project is completed with Installation, Uninstallation, Registration, Login, Reset Password, MySite activities working correctly. We still need to give the admin the ability to send mass emails for the Dashboard, although individual sending admin abilities... We did not know if you wanted a selective mass email, or just an email to send to everyone signed up on Quartz. So we wrote the function to send an email to everyone on the list. We also had trouble on getting the picture to auto crop, this proved to be

Projected Schedule from First Submission:



We did not have much time for beta testing and actually spent most of the time coding until the end because many of our activities did not pass during our team meetings.

Our Project CheckLists (Items that we completed have checks):

a) Account Creation –

- check submission function ✓
- check error messages displaying relevant errors ✓
- check if all textboxes are present ✓
- check if the system recognizes correct id and password, and blocks maleficent accounts ✓

b) Uninstalling Quartz –

- check if the system generates emails as test cases indicate
- check if the system unauthorize the user for uninstallation for wrong id and password - check if the user can safely uninstall the application
- check if the admin receives an warning email that user has tried incorrect id and password after a number of tries
- check if the user is correctly deleted in the database

c) Course Editing -

- check if user can generate emails once the server is back on, and let the admin know the email
- check if the user can create new courses ✓
- check if the user can delete existing courses ✓
- check if the user can enter dates using combobox
- check if the users can receive emails from his or her department
- check if there are error checkings when deleting items ✓
- check if the cancel button correctly resets the form ✓
- check if the user can edit his or her website without the browser refreshing if the admin - check if admin can retrieve information when the server fails

d) Installing Quartz

- check if the user can download Quartz.zip ✓
- check if the user can install Quartz application ✓
- check if the system generates correct URL for startupv3.html, depending on the platform (e.g, if user has mac or pc) ✓
- check if the help button works✓
- check if the slash at the end of URL cause any problems - check if the correct guide messages pop up ✓

e)Resetting Password

- check if we can generating a php file (such as the log of recent activities). ✓
- check if we can generate a html file from the php file✓
- check if we can generate a different type of file if the html file does not work(word, excel) ✓

f) Admin Functions

- check if the database will have the ability to update itself -check if admin can be granted ✓
- check if admin privileges can be revoked
- check if user can update new software of Quartz ✓
- check if admin can manually activate a user's account ✓

Project Quartz Documentation:

Eight Software Questions:

- 1) The kind of system that the user is dealing with databases.
- 2) The user will install the system by downloading a file and then clicking on the install.php file within Quartz.
- 3) The system allows the user to create accounts, reset their passwords, edit their manage pages.
- 4) The overall structure of the system is that of an Model-View-Controller(MVC) structure. The model is encompasses model stubs, the view is just the CSS and html, and the controller is the framework. The system takes user input and then uses that to update the model. Then the controller takes what is inside the model(just holds information) and updates the view (what the user sees).
- 5) The components consist of individual activity classes and which are run by the definition classes load the page/view.
- 6) The interface to each component are the activity definitions classes.
- 7) The data that are involved are the hard coded data for the model stubs for some of our activity classes and the user's input. This data is organized by methods and if statements that captures user's input and based on the user's input the system will process the data through a for loop, thus showing the page.
- 8) The main methods, which are the show methods, just show the form based on a multiple of many other methods.

Issue/Solutions Statements:

Controller:

Issue: How does the controller know which text fields correspond to which variables?

Solution: Each text field has an id assigned so the controller knows what to store in each variable.

Issue: How does the controller change the view based on when the users press the buttons?

Solution: It changes the coordinates of the geese image by subtracting and adding fixed numbers depending on what keys are pressed(left or right).

Issue: How does the controller print out how many geese are needed?

Solution: A for loop based on the user's input is used to print out all the geese.

View:

Issue: How is the background created?

Solution: Java graphics is used to draw the necessary shapes in the picture.

Issue: How do the picture images get stored?

Solution: The images are stored with a javascript function called getImage data which stores the pictures as snapshots.

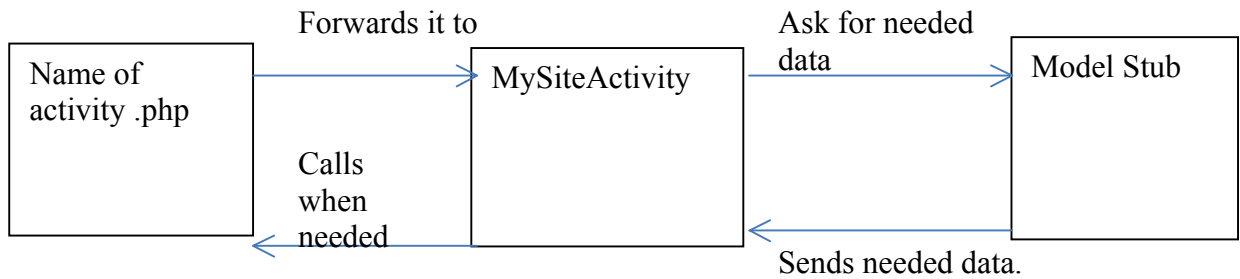
Model:

Issue: How does user input get stored?

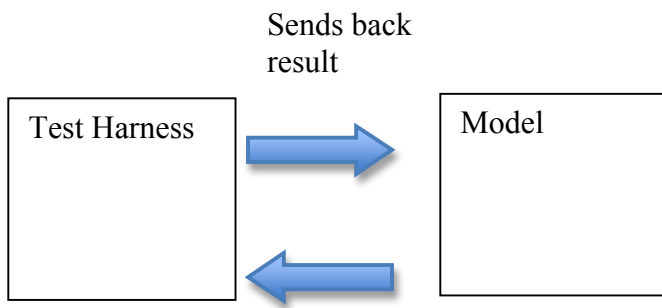
Solution: The user input is stored into variables by the information from the variables' id tags.

Diagram:

Framework:



Api :



Test out API
functions.

Appendix

Group Status Report.....	1
Gant Chart.....	2
Project Checklist.....	3
Project Checklist.....	4
Eight Software Questions.....	5
Issue/Solution Statements.....	6
Diagram	7