Intro To Computer Science
Guild Programming

Semester Project

Instructor: William A Loring

Table of Contents

The semester project will provide your team with hands-on experience in applying your programming and problem-solving skills to create a functional and user-friendly system. The project will provide a practical application of programming concepts learned during the course. You will work in teams to create a functional system and present your final product at the end of the semester.

Agile Software Development

Agile development is the current and most common process for software engineering and application development activities.

What is Agile?

Agile development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams. Agile methods or Agile processes generally promote a disciplined project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices intended to allow for rapid delivery of high-quality software, and a business approach that aligns development with customer needs and company goals.

https://www.cprime.com/resources/what-is-agile-what-is-scrum

DRY Software Engineering

Don't **R**epeat **Y**ourself (**DRY**) is a principle of software engineering aimed at reducing repetition of software patterns. It you are repeating any code, there is probably a better solution.

Violations of DRY are typically referred to as WET solutions, which is commonly taken to stand for "write every time", "write everything twice", "we enjoy typing" or "waste everyone's time".

Create classes with a single purpose or theme. Abstract as much as possible from the application to the classes. The application contains as little logic as possible. It creates and uses objects and their methods.

Guild Based Software Engineering

Software engineering is rarely done alone. It is almost always done as part of a team.

The class will be divided into guilds of 2-3 people each. Guild membership is assigned by the Guild Master (the instructor).

Share Your Experience

In any class, some students, depending upon their major, programming experience, artistic talent, etc. may exhibit more proficiency than others on certain aspects of the assigned course work. A Guild gives an opportunity for everyone on the team to work together and share their experience.

This project is as much about working as a team as it is about the assignment. The process of coming together, helping each other out is a huge part of team-based work.

Guild Assignments

Guild assignments are team projects where everyone works on the same code base.

Individual assignments are not team projects. Everyone does their own work in an individual assignment.

If a Guild member does not respond to communication within 24 hours, move on without them.

Outlook Groups

An Outlook Group will be created for each Guild. This can used to share documents, send email, and work collaboratively.

Each Team has a similar Outlook Group Email address. The last number is the only thing that is different.

For example, if you are in Group 1: Loring-Java-Group01

Please watch this video on how to use Outlook Groups. https://youtu.be/DDG5n0QdvUg

How to Work with Word in Outlook Groups

Individual Grade

YOUR GRADE for GUILD PROJECTS IS FROM THESE THREE ITEMS.

- Individual Evaluations
- The result of your Guild weekly sprint
- Communication

Guild Software Engineering Process

This is a highly suggested process that is like working in real world Agile software engineering.

GitHub Repository

Each Guild will have a shared GitHub Repository. This repository is used to store any code or text documentation for the project.

Development Process

Get Started

- Live scrum (meeting) to create pseudocode.
  - The pseudocode is a text file.
- Divide up coding into small tasks. Put into the weekly KanBan board.
  - Read **GitHub KanBan Board** in Resources in Blackboard.

Development

- Comment your code descriptively.
  - Why did you do this, is there something left to do, did you hard code something for testing,
- Commit small changes early and often.
- Leave detailed commit descriptions.
  - Make sure the code compiles before pushing to GitHub.

Testing

- Test the code to ensure it meets the requirements.

Conclusion

- Briefly describe who did what.
- Briefly describe your Guild’s process.

- Each week/sprint will result in a working application.

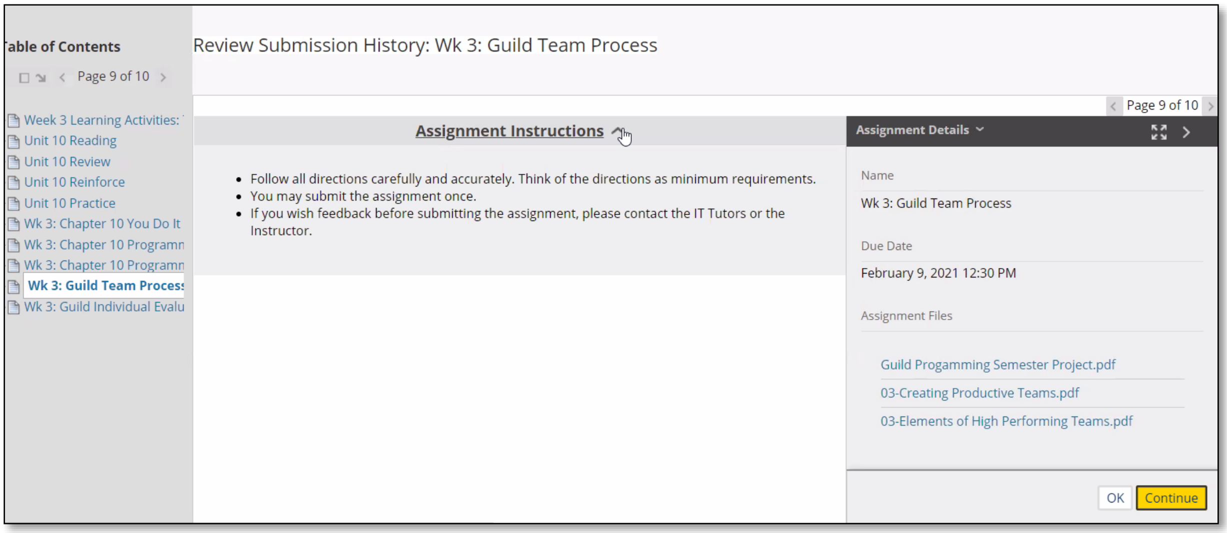
Assignment Minimum Requirements

Think of this document as setting out the minimum requirements for the project. If the minimum requirements are met in your assignments, you have license to be creative and go further than the assignment requires.

Guild Team Assignment Instructions

The Guild Team assignments are a bit different than the individual assignments.

1. Go to the **Guild Team** Assignment.
2. Click **Assignment Instructions** to see the general instructions.
3. Click **Assignment Details** to see the pdfs that are attached to the assignment.



Review Submission History: Wk 3: Guild Team Process

- Follow all directions carefully and accurately. Think of the directions as minimum requirements.
- You may submit the assignment once.
- If you wish feedback before submitting the assignment, please contact the IT Tutors or the Instructor.

Name
Wk 3: Guild Team Process

Due Date
February 9, 2021 12:30 PM

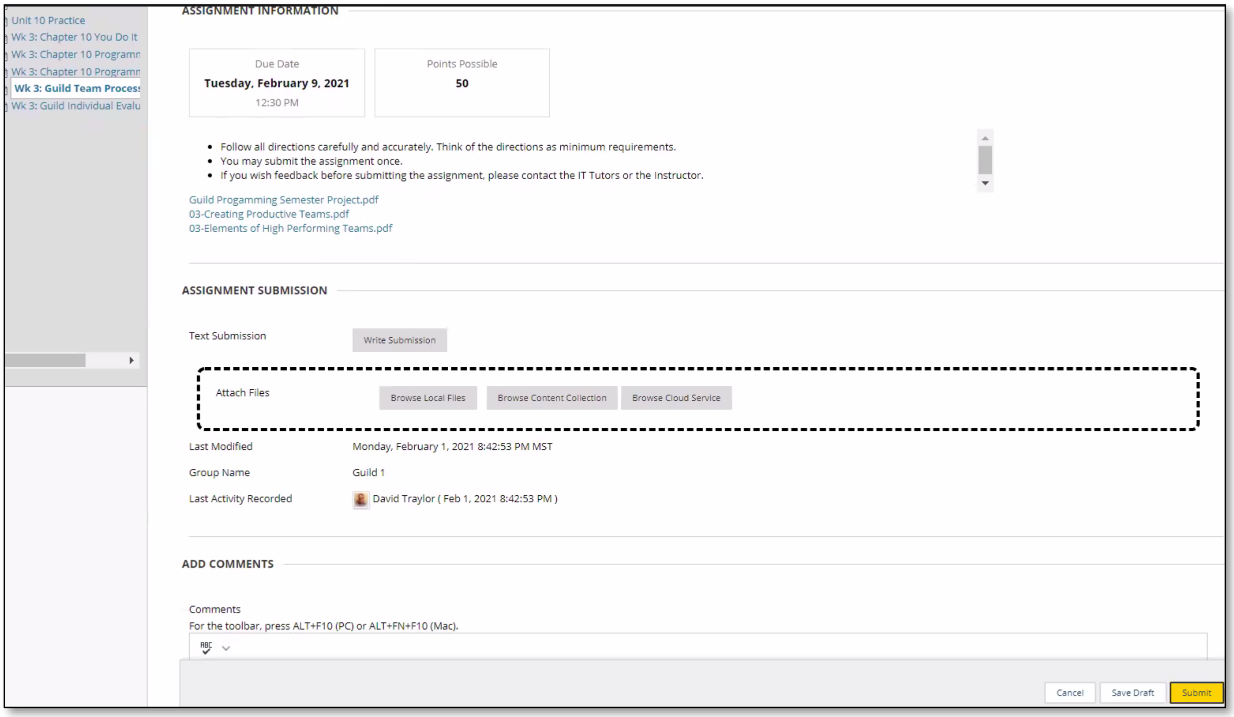
Assignment Files

Guild Progamming Semester Project.pdf
03-Creating Productive Teams.pdf
03-Elements of High Performing Teams.pdf

Guild Team Assignment Submission

The current Guild leader is responsible for submitting the Guild assignment.

1. Go to the **Guild Team** Assignment.
2. Click **Continue**.
3. You will see the standard submission area.
4. Attach the requested assignment submission and Click **Submit**.


Unit 10 Practice
Wk 3: Chapter 10 You Do It
Wk 3: Chapter 10 Programn
Wk 3: Chapter 10 Programn
Wk 3: Guild Team Process
Wk 3: Guild Individual Evalu
Write Submission
Browse Local Files
Browse Content Collection
Browse Cloud Service
Cancel
Save Draft
Submit

ASSIGNMENT INFORMATION

Due Date: **Tuesday, February 9, 2021** 12:30 PM

Points Possible: **50**

- Follow all directions carefully and accurately. Think of the directions as minimum requirements.
- You may submit the assignment once.
- If you wish feedback before submitting the assignment, please contact the IT Tutors or the Instructor.

Guild Progamming Semester Project.pdf
03-Creating Productive Teams.pdf
03-Elements of High Performing Teams.pdf

ASSIGNMENT SUBMISSION

Text Submission

Attach Files

Last Modified: Monday, February 1, 2021 8:42:53 PM MST

Group Name: Guild 1

Last Activity Recorded: David Traylor (Feb 1, 2021 8:42:53 PM)

ADD COMMENTS

Comments
For the toolbar, press ALT+F10 (PC) or ALT+FN+F10 (Mac).

Guild Shared Software Engineering GitHub Process

Many of you are brand new to GitHub. We are going to use a basic approach. It requires communication with the team (which isn't a bad thing!).

Organize

- A divide-and-conquer approach is a good way to work on these assignments. When pair programming and working asynchronously, the number 1 thing is to organize the work first so people aren't writing duplicate methods. Even if you are perfectly committing, pushing, and pulling, you'll end up tripping over each other's code.
- Pens shouldn't touch paper until everyone knows exactly who is doing what.

Commit

- Commits should be small and frequent. Add a new method? Commit. Add a new data type? Commit. Have comments to add? Commit.
- Your commit messages should be written to not just say what you did, but why you took an approach. Treat this as your voice to the team, so they almost don't need to talk to you to see why you did something.

- The origin (GitHub) should be the source of truth for your work. If your local work is different, it needs to be committed or overridden with the origin since it's out of date.

Before You Commit

- **Pull**. You want to be sure that you're committing to the most recent version of the branch. Pulling right before committing will help avoid potential merge conflicts.

When You Commit

- **Push** immediately. That ensures that the origin (GitHub) is perfectly in sync with the most recent changes for everyone else. Remember: your commits are for everyone else so they're up to date with the current code.
- Message your group and let them know that you pushed. Yes, we should be pulling before committing, but it's a courtesy and avoids any potential issues.
- If you pull and there are conflicts that need to be resolved. . .
  - Say you pull and your teammate made some changes on that file. The easiest way to resolve this is to first copy your changes somewhere else (these should be small if we're committing correctly)
  - Accept the changes from the pull (Remember GitHub is the source of truth)
  - Add your changes back in
  - Commit then push.

Example Project Workflow

Here is an example of project workflow. You and I are on the same team on this project.

First

- We meet, talk over the pseudo, and setup a plan. You take conversion methods and I take display methods or something.
- Either setup the main class together or whoever can do it right away volunteers. We set it the main class together and push it.

Workflow

- Pull and double-check I'm up to date.

- I write a small method displayInches()
- I pull again to check for changes
- Commit my changes. My commit comment says:
    - "Add displayInches method. Right now, it's hard-coded since we don't have a conversion yet but will update once we get data".
- Push
- I send message to the team:
    - Hi everyone I just pushed to add displayInches()"
- Repeat

Your combined code is going to have different styles and approaches. That's perfectly okay and expected. When you go and work in the field, you will see a multitude of different styles and approaches in large codebases and will have to get used to that. If someone doesn't agree with a solution or approach, discuss it, and go with what you collectively decide.

Approach the project with the mindset that it is a single application and not a single school assignment: 1 project 1 team. These discussions can be one of the bigger challenges of development and something you will regularly do day-to-day.

Shared Coding Process

NOTE: All coding is done locally and committed to GitHub.

1. Your Guild meets in real time.
2. Read the program requirements.
3. Create pseudocode KanBan.

Pair Programming

1. Everyone works on the pseudocode in real time.
2. One person is the driver, the rest are navigators.
3. The driver does the writing of the code.
4. The navigators provide input, look up resources.
5. Divide up the project into parts.

6. Create the basic files and shell of the program

Asynchronous Programming

This is an example of dividing up the coding project.

1. One person declares the variables and input.
2. One person does the calculations.
3. One person does the display.

Week 7 Milestone: Guild Project Kickoff and Team Charter

100 points

Time required: 90 minutes

1. **Read:** Creating Productive Teams
2. **Read:** Elements of a High Performing Team
3. **Read:** Team Charter

Team Charter

Do: Determine how your Guild will handle the following items. This will be your team charter.

1. What is the name of your Guild?
    a. The Intro to Computer Science Guild Project repository will be name with your Guild Name. Please indicate which Guild it is, Guild 1, etc.
2. Communication
    a. Synchronous
        i. Zoom
        ii. Teams
        iii. Skype
        iv. Video conference software of your choice
    b. Asynchronous
        i. Outlook Group Email

3. Collaboration on documents
    a. GitHub
4. The Guild will create a list of the skills needed for the project.
    a. Team skills
    b. Computer skills
5. Each member will create a list of the skills they bring to the project.
    a. Team skills
    b. Computer skills
6. Guild leader of the week
    a. The Guild leader is not responsible for doing all the work, only organizing, or getting everyone together.
    b. Rotate between each member for Guild leader of the week.
    c. A recommended practice is to set a schedule for Guild leader rotation at the beginning of the project.
    d. The Guild leader is responsible for submitting the project assignment for that week.

Do: Determine how your Guild will handle the following items.

1. What is the name of your Guild?
2. Communication
    a. Synchronous (required)
        i. Skype
        ii. Zoom
        iii. Video conference software of your choice
    b. Asynchronous
        i. Outlook Group Email (Required)
3. Collaboration on documents
    a. Outlook Group Files

4. The Guild will create a list of the skills needed for the project.
    a. Team skills
    b. Computer skills
5. Each member will create a list of the skills they bring to the project.
    a. Team skills
    b. Computer skills
6. Guild leader of the week
    a. The Guild leader is not responsible for doing all the work, only organizing, or getting everyone together.
    b. Rotate between each member for Guild leader of the week.
    c. A recommended practice is to set a schedule for Guild leader rotation at the beginning of the project.
    d. The Guild leader is responsible for submitting the project assignment for that week.
7. Expectations for Guild Members
    a. Set your shared expectations for the Guild
    b. Responsiveness
        i. What are the expectations for response to communication?
    c. This is a collaborative team project.
        i. It is not ok to say to the rest of the team: plan this week without me and let me know what I am supposed to do.
        ii. Collaboration means each team member contributes equally.
    d. Deadlines for individual assignments
    e. Deadline for team assignments
    f. If these expectations are not met, ask the team member why
    g. If there is no response, move on without them
    h. Give an honest evaluation of individual performance with the evaluation

Assignment submission

Please submit your Team Charter to the shared assignment.

Shared Code Development with GitHub

This assignment will take your Guild through a tutorial about working with Github in a shared development environment. This will prepare you for the first shared coding assignment at the end of this tutorial.

Each member of the Guild completes these steps.

1. Click the link to the assignment. https://classroom.github.com/a/OLn0YFq3
2. Join the classroom: **Intro to Computer Science Fa24.**
3. Select your name.
4. Click **Join** by your Team Name.
5. Click on the assignment repository link.

Code Commits with GitHub Desktop or Visual Studio Code

- Git Going with GitHub Tutorial (GitHub Desktop)
- Visual Studio Code Git Tutorial

It is much easier to work with code on your local pc with your own editor. Use GitHub Desktop or Visual Studio Code to make and commit changes to the assignment repository.

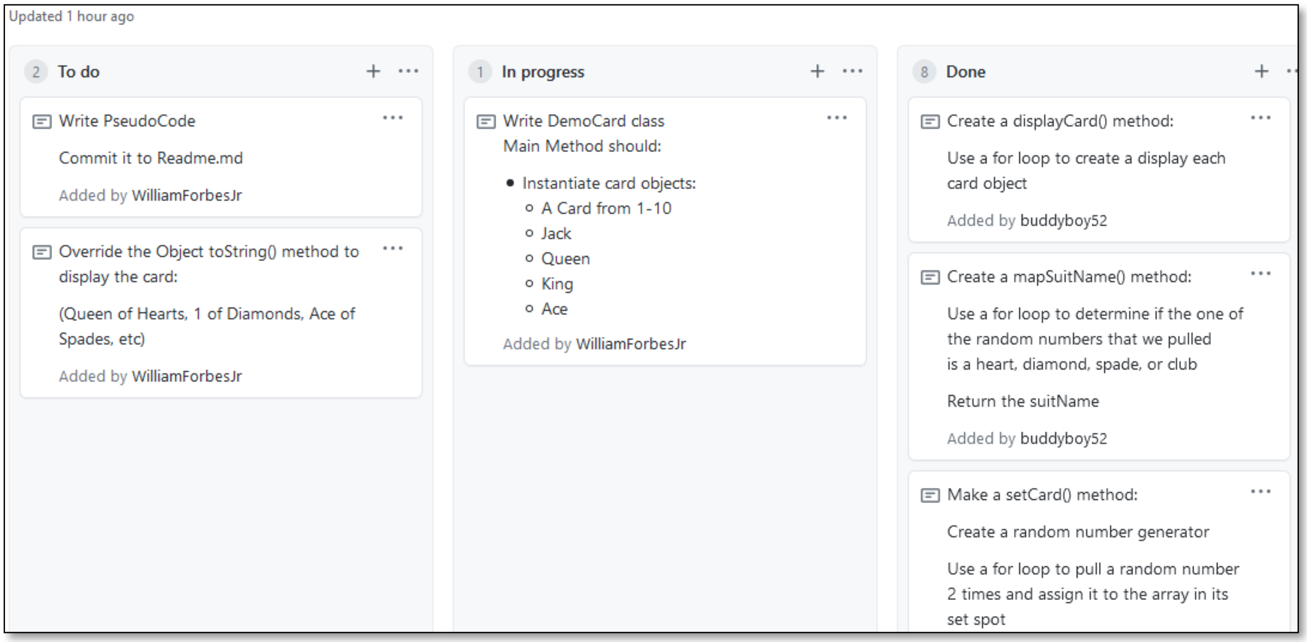
1. Go to the Guild assignment repository. https://classroom.github.com/a/OLn0YFq3
2. Clone or Pull the repository to your local computer. This synchronizes the GitHub repository to your local computer code repository.
3. Personalize your readme.md with your name and information about your project.

Shared KanBan Board

1. Create a shared KanBan (Project) board.
2. Each member will add an item.

GitHub KanBan Board Tutorial

Sample Kanban board.



Updated 1 hour ago

2 To do

Write PseudoCode

Commit it to Readme.md

Added by WilliamForbesJr

Override the Object toString() method to display the card:

(Queen of Hearts, 1 of Diamonds, Ace of Spades, etc)

Added by WilliamForbesJr

1 In progress

Write DemoCard class
Main Method should:

- Instantiate card objects:
  - A Card from 1-10
  - Jack
  - Queen
  - King
  - Ace

Added by WilliamForbesJr

8 Done

Create a displayCard() method:

Use a for loop to create a display each card object

Added by buddyboy52

Create a mapSuitName() method:

Use a for loop to determine if the one of the random numbers that we pulled is a heart, diamond, spade, or club

Return the suitName

Added by buddyboy52

Make a setCard() method:

Create a random number generator

Use a for loop to pull a random number 2 times and assign it to the array in its set spot

Assignment Submission

1. **Guild GitHub Assignment:** The Guild coding assignment is coded and submitted in GitHub.
2. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.
3. **Guild Individual Evaluation:** Each Guild member submits a Guild Individual Evaluation in Blackboard.

Week 8 Milestone: Planning and Pseudocode

100 points

Time Required: 60 minutes

NOTE: Each person contributing to the planning:

1. Highlight your text in a different color.
2. Put your name by the parts you worked on.

This week is all about planning and high-level pseudocode. No coding this week.

Develop your project idea into a real project.

Watch: What's an algorithm? - David J. Malan

Watch: BBC Learning - What Is an Algorithm

Choose and Plan Your Project

As a team, brainstorm a project, backstory, wireframe, and some starting pseudocode. Use your readme.md as a place to organize your project narrative.

There are project ideas at the end of this document. Develop your projects as shown in these examples. You can program in Python, Java, C++, or any combination.

Use the attached project planning outline to plan your project.

Program Requirements

- Pseudocode and Weekly KanBan board first.
    - Use Weekly KanBan for individual assignments
    - You can use TODO's in the code.
- Create a Weekly folder in the repository for each coding sprint.

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 9, Milestone 10, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild coding assignment is coded and submitted in GitHub.
3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.
4. **Guild Individual Evaluation:** Each Guild member submits a Guild Individual Evaluation in Blackboard.

Week 9 Milestone

Shared Coding Process

1. Pseudocode first
2. Split up coding tasks
3. KanBan

4. Code and communicate
5. Commit often
6. Test and submit

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 9, Milestone 10, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild coding assignment is coded and submitted in GitHub.
3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.
4. **Guild Individual Evaluation:** Each Guild member submits a Guild Individual Evaluation in Blackboard.

Week 10 Milestone

Shared Coding Process

1. Pseudocode first
2. Split up coding tasks
3. KanBan
4. Code and communicate
5. Commit often
6. Test and submit

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 10, Milestone 11, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is created in GitHub.

3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.

4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.

Week 11 Milestone

Shared Coding Process

1. Pseudocode first
2. Split up coding tasks
3. KanBan
4. Code and communicate
5. Commit often
6. Test and submit

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 8, Milestone 9, etc. That will give you a way to easily go back to a known working version if the next version has trouble.

2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is created in GitHub.

3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.

4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.

Week 12 Milestone

Shared Coding Process

1. Pseudocode first
2. Split up coding tasks

3. KanBan
4. Code, commit, communicate
5. Commit often
6. Test and submit

Program Requirements

- Don’t do everything at once.
- You do not have to do this is one week.
- Pick what you can do in a week and get that to work. That is Agile development.

Assignment Submission

Part of the assignment evaluation is the quality of planning, pseudocode, committing, and KanBan board use.

Each sprint (week) should have a functioning program. That is Agile development.

1. **Milestone Folder:** Create a Milestone folder for each week’s successful iteration of your project. Milestone 8, Milestone 9, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is created in GitHub.
3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.
4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.

Week 13 Milestone

Add a new functional feature to your program.

Shared Coding Process

1. Pseudocode first
2. Split up coding tasks
3. KanBan

4. Code and communicate
5. Commit often
6. Test and submit

Program Requirements

- Don’t do everything at once.
- You do not have to do everything in one week.
- Pick what you can do in a week and get that to work. Pick something else next week.
-

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week’s successful iteration of your project. Milestone 8, Milestone 9, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is created in GitHub.
3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.
4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.

Week 15 Milestone

Add a new functional feature to your program.

Finish Up Program

1. Complete and polish up your program.
2. Complete any features.
3. Comment and organize your code as if someone else will be taking over your project.

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 8, Milestone 9, etc. That will give you a way to easily go back to a known working version is the next version has trouble.

2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is created in GitHub.

3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.

4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.

Week 16 Milestone – Project Presentation

Please read all the directions before beginning the assignment.

Requirements

Work on completing any features that are not completed.

Comment and organize the code.

Presentation Requirements

This will be a screencast with each member presenting a part of the program.

1. Teams will present their systems to the class.

2. Describe how you developed your program.

3. Demonstrate the flow of the program.

4. Discuss challenges faced, solutions implemented, and the impact on the fictional customer's business.

5. What worked and what didn't work.

Program Requirements

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 8, Milestone 9, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is created in GitHub.
3. **Guild Team Submission in Blackboard:** The Guild leader submits a screenshot of their GitHub repository.
4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.

Finals Week: Project Completion

Please read all the directions before beginning the assignment.

Final submission of your application.

Requirements

1. Comment each line of code as shown in the tutorials and other code examples.
2. Follow all directions carefully and accurately.
3. Think of the directions as minimum requirements.

Program Requirements

1. A fully functioning application.
2. Comment and organize your code as if someone else is going to be working on it.

Assignment Submission

1. **Milestone Folder:** Create a Milestone folder for each week's successful iteration of your project. Milestone 8, Milestone 9, etc. That will give you a way to easily go back to a known working version if the next version has trouble.
2. **Guild GitHub Assignment:** The Guild pseudocode, KanBan board, and code is stored in GitHub.

3. **Guild Team Submission in Blackboard:** Submit a link to a short screencast showing the functionality of the program.
4. **Guild Individual Evaluation:** Each Guild member submits an Individual Guild Evaluation in Blackboard.