# Team Project CEN 4010

## 120 points

You will be creating an online web application bookstore which targets a particular niche in technology. The application, named **Geek Text** (think of thinkgeek meets barnes and noble) will need to support the following features:

| Feature ID | Feature | Benefit |
|---|---|---|
| 1 | Book Browsing and Sorting | Users will have a simple and enjoyable way to discover new books and Authors and sort results.<br><br>DESCRIPTION: Allow user to browse books by genre, top sellers in our book store, and book rating with pagination based on 10 or 20 results. Allow Sort by book title, author, price, book rating, and release date. |
| 2 | Profile Management | Users can create and maintain their profiles rather than enter in their information each time they order<br><br>DESCRIPTION: Users can manage their login credentials (ID, password), personal information (name, email address, home address), nickname for book rating and commenting, credit card information (multiple), and shipping address (multiple). Physical addresses, email addresses, and credit card info should be verified as valid. Passwords must meet our current security standards |
| 3 | Shopping Cart | Users can manage items in a shopping cart for immediate or future Purchase<br><br>DESCRIPTION: Users can easily access their cart from any page, view the same information displayed in the book list, change the quantity, remove it from their cart, or save it for later. A subtotal for all items in their shopping cart should be displayed at the bottom. Items saved for later should appear below that. |
| 4 | Book Details | Users can see informative and enticing details about a book<br><br>DESCRIPTION: Display book name, book cover (which can be enlarged when clicked), author and bio, book description, genre, publishing info (publisher, release date, etc.), book rating, and comments. Hyperlink author's name to a list of other books by the same author. |
| 5 | Book Rating and Commenting | Users can rate AND comment on books they've purchased to help others in their selection<br><br>DESCRIPTION: For Rating: Use a five-star rating system. Users can only rate a book if they've purchased it, and may select whether they show their nickname (defined in their profile) or remain anonymous. For Commenting: A single comment should be limited to the number of characters, which can fit within half the browser window (so that there are at least two comments which can appear at the same time). Users can only comment on a book if they've purchased it, and may select whether they show their nickname (defined in their profile) or remain anonymous. |

Each feature will be owned by a team member. This means that although you will earn an overall team grade, each person in the team will be responsible for a feature being done. If the feature is no completed, the individual who owns the feature will not be eligible for a group grade.

## How will the group develop the web application?

The team will use scrum, which is an agile methodology used to develop software. You will need to review Lecture 1 Scrum to become familiar, but here is a summary:

The team will develop the application in timeboxed intervals known as Sprints. Each **Sprint** has a start date and an end date. At the beginning of the sprint, the team conducts a **sprint planning** meeting, in which they discuss and estimate what is the functionality that they will implement during that Sprint. The functionality will be implemented by breaking down the features into **user stories**. A user story is a description what the software must do, the benefit and the acceptance criteria. During the sprint, the team will meet through a meeting known as the **standup or daily scrum**, in which each team member mentions what they are working on, what they have accomplished and any impediments. It is expected that the team does at least 2 standup meetings each week. Finally, after the sprint has finished, the team hosts a **Sprint Review**, a meeting in which they demo (show) the work that was completed during the sprint. After the Review, the team performs a **Sprint Retrospective (Retro)** in which they discuss what has gone well and what can be improved. It's a chance for the team to incorporate improvements for the next Sprint. After completing the Sprint Retrospective, they will then schedule the sprint planning meeting for the next sprint and the process starts again.

### Roles:

The teams will consist of 5 people which will be the following during each sprint:

**3 Developers:** The developers will be responsible for creating their feature and writing their own users stories.

**1 Product Owner:** This will be an alternating role. This person will review the backlog of all the user stories written by developers and make sure they meet the requirements of the features. They will be responsible for asking any questions for that sprint in regards to product functionality. This will be a shared role as it is expected that they also pull in some stories for development

**1 Scrum Master:** The scrum master will be the lead person who oversees removing any impediments and bring up any issues to me during the sprint. This will be a shared role as it is expected that they also pull in some stories for development

The team will have a different Product Owner and Scrum Master in each sprint, so everyone will have an opportunity to work under each role.

### Velocity:

Velocity is the measurement of how much work will be done. We will use hour units to estimate our work. So, if a person is working on a given user story that will take them 6 hours to achieve, this will be the estimated effort of their user story:

A User Story is a description of the development task that has to be done, so the teams will have multiple user stories and hour estimate to each user story.

Each team will calculate their capacity (how much work they can do) for the sprint in the following manner:

Each developer is expected to work 4 hours a week, so each developer will have an 8-hour allocation per Sprint.

So, 5 developers * 8 = 40 combined hours of work for each a team per Sprint.

The product owner during the sprint will be in charging of grooming the backlog and breaking down the features into user stories.

### Timeline:

The project will be completed in 6 sprints, each lasting 2 weeks. Sprint 1 will start on the 3rd week of class.

Date of Sprint Reviews:

- Sprint 1 1/22 - 2/2 Sprint Review 2/3 (Record your demo and post in your folder)
- Sprint 2 2/4 – 2/16 Sprint Review 2/17 (Live Demo on Google Hangouts)
- Sprint 3 2/18- 3/2 Sprint Review 3/3 (Record your review and post in your folder)
- Sprint 4 3/4- 3/23 Sprint Review 3/24 (Live Demo on Google Hangouts)
- Sprint 5 3/25- 4/6 Sprint Review 4/7 (Record your demo and post in your folder)
- Sprint 6 4/8-4/20 FINAL Sprint Review 4/21 (Live Demo on Google Hangouts)

The Live Sprints reviews will be done via Google Hangouts on Saturdays from 9am to 12pm (Check the Calendar provided in this document). All team members need to attend the reviews.

The recorded reviews will be done by the team and will need to be posted in their team folder by Sunday 6pm.

You can use the software of your choice to record the review as long as the output is an mpeg. Screencastomatic (https://screencast-o-matic.com/) is free to use for recordings.

## Deliverables:

There will be multiple deliverables for this project, but what I value most is a working project which will be demoed to me though online meetings. When the project is completed, you will have.

1. Source Code for Project in GitHub
2. Documents for Scrum Ceremonies
3. Recordings of Reviews (if they cannot be done live)
4. Final Demo completed (Will happen after Sprint 6 is completed)

The source code should be checked into Github. Please make sure everyone has an account.

## Scrum Documents:

During each Sprint, each team will perform the following scrum ceremonies:

- **Daily Standup:** A quick 5-minute meeting in which the team members discuss what they accomplished the since they last met and what they plan to work on. Also, any impediments are brought up. You are required to have 2 standups every week. The team can determine what day and time they will meet.
- **Sprint Planning:** A meeting to determine which of the backlog items will get pulled into the upcoming sprint. This document will keep the details of the user stories that will be worked on during the sprint.
- **Sprint Review:** A meeting to showcase the completed stories to the stakeholders. Depending on the sprint, we will either do these live via google hangouts (Sprints 2,4,6) or you will record them (Sprints 1,3,5).
- **Sprint Retrospective:** A Meeting to determine what went right during the sprint and what can be improved.

During each ceremony, the scrum master will keep notes and load them into the Google Drive under the appropriate folder per team:

- ➔ Team 1
  - Sprint 1
    - StandupNotes document
    - SprintPlanning document
    - SprintReview document
    - SprintRetro document
  - Sprint 2…
- ➔ Team 2

- Sprint 1…..

The Public Google Drive for Class is:

https://drive.google.com/open?id=1SocRh2dhZORPmm56HhTwNRkBjzMZXWHL

In this drive I will place the document templates for the teams to use.

The scrum master in each sprint must download the templates, create a folder for the sprint and upload the documents for every sprint.

For examples of how a completed document would look, please refer to the Examples folder in the google drive. Here I have examples of the scrum documents and recorded demos.

**IMPORTANT:** I will grade the documents on a sprint by sprint basis on Sundays at 6pm. All Sprint documents must be completed by SUNDAY 6pm for the Sprint that has ended.

For example, If Sprint 2 ends on 2/16 and the view takes place on 2/17 (Saturday) then all documents must be completed by 2/18 Sunday at 6pm.

## Grading Criteria Checklist:

**120 points:** Exceptional project. All scrum ceremony documents are completed. All reviews have fully demonstrated working code.

**100+ points: Good project:** Great project but can be improved upon. Some ceremony documents are missing. Some of the features are not fully functioning.

**80+ points: Average project:** The project does is missing the implementation of some of its core feature. Ceremony documents are incomplete.

**40+ points:** Barely any working software demonstrated, just mostly nonfunctional code. Hardly any documentation completed.

**Instructor Grading Criteria:**

**Software Implementation (50 points)**

- Profile Management ____ (3 very poor, 5 average, 7 above average, excels 10 )
- Book Browsing and Sorting ____ (3 very poor, 5 average, 7 above average, excels 10 )
- Book Details ____ (3 very poor, 5 average, 7 above average, excels 10 )
- Shopping Cart ____ (3 very poor, 5 average, 7 above average, excels 10 )
- Book Rating and Commenting ____ (3 very poor, 5 average, 7 above average, excels 10 )

**Scrum Execution (60 points)**

Sprint 1 Ceremonies in Folder ____ (3 very poor, 5 average, 8 above average, excels 10 )

Sprint 2 Ceremonies in Folder ____ (3 very poor, 5 average, 8 above average, excels 10 )

Sprint 3 Ceremonies in Folder ____ (3 very poor, 5 average, 8 above average, excels 10 )

Sprint 4 Ceremonies in Folder ____ (3 very poor, 5 average, 8 above average, excels 10 )

Sprint 5 Ceremonies in Folder ____ (3 very poor, 5 average, 8 above average, excels 10 )

Sprint 6 Ceremonies in Folder ____ (3 very poor, 5 average, 8 above average, excels 10 )

**Working Code Deployed**

Uploaded to GitHub ____ (0 no, 10 yes)

## Document Templates:

The document templates are in the template folder in the google drive, make a copy of them to use in your sprints.

**TeamStandup:** This document should have 1 entry per class (or twice a week) where team members discuss what they are working on until the project is completed.

**TeamPlanning:** This document is filled out once per sprint at the beginning and will have the stories pulled into the sprint, their estimate and who is assigned to them. The team also agrees to a Sprint Goal (what do they plan to accomplish in the sprint).

**TeamRetro:** This document is filled out once per sprint after the review and will have 1) what went well 2) what didn't go well and 3) what improvements can be made.

**TeamReview:** This document is filled out before the review occurs and WORKING software is demoed to the product owner or stakeholders. The product owner should review which stories were completed and if any, were not completed.

## Working with User Stories:

The following are example stories have been provided to understand how to write the user stories:

Feature: **Profile Management**

User Story Format Example:

Title

As a book browser or purchaser, I can create a user profile So that I do not have to enter my information each time I add books to my shopping cart or purchase books

Acceptance Criteria:

Functional Requirements (required)

User can enter a user name which will stored on a database
User can enter a first name and last name which will be stored on a database
User can enter a password which will be stored on a database
User can enter an address which will be stored on a database
User will click on a save button to persist the info on a database

NonFunctional Requirements (optional)

When the user clicks on their profile, the profile information should load in less than 2 seconds.

Estimate
8 (hours)

Our planning document template only has the following info

**StoryId, Title, Status (New, Implementing, Done ), AssignedTo , Estimate , Acceptance Criteria.**

## Final Demo

During the final weeks of class, the teams will have the project ready to demo from a given sandbox. I will ask you to demo the core features and any additional features.

I will provide all the Sprint Review Schedules once the teams are assigned. Please refer to the following Google Calendar:

**FIU Roque CEN 4010**

https://calendar.google.com/calendar/embed?src=1pthsodlk0craqsc992pmp5g1s%40group.calendar.google.com&ctz=America/New_York

## Getting Started Checklist:

Before Sprint 1, you need to have the following ready:

1. Meet your team members and exchange contact.
2. Review Lecture 1 to understand the scrum fundamentals.
3. Agree on technology and architecture
4. Decide for Sprint 1 who would be:
    a. Product Owner
    b. Scrum master
5. Review Document Templates
6. Review User Story example
7. Make sure you can access Google Drive
8. Make sure everyone has a working webcam and mic
9. Create GitHub project for your team

To assure that the group has a clear understanding of the items in this checklist, there will be a project kickoff meeting scheduled for each group. Please Check the calendar for the date.