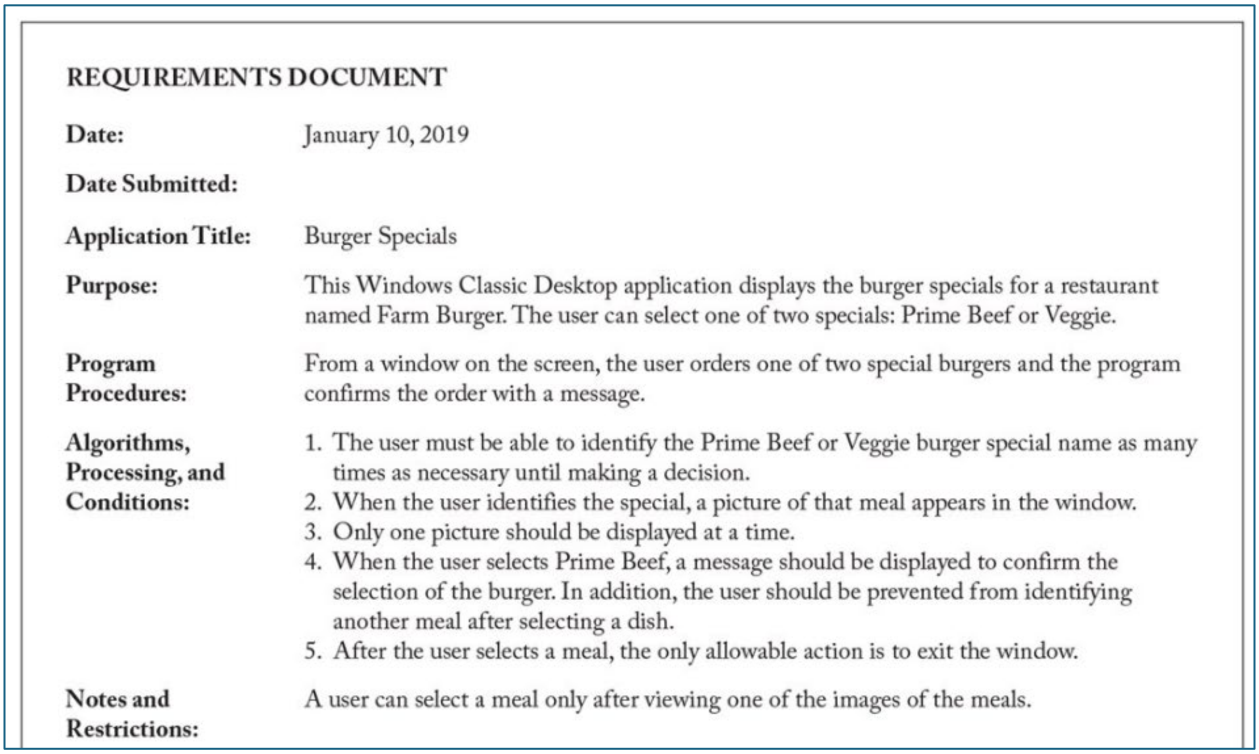
GUI Layout w/Event Plan (A3) and Working Application (A4) – 40 points per assignment

This assignment will be delivered in two parts, each worth 40 points. Assignment A3 is due Monday 2/12; Assignment A4 is due Monday 2/19. Deliverables are discussed below.

This assignment will let you practice laying out a GUI to specifications, creating an event plan from a problem definition, and coding a working GUI-based application. Code tools should either be Visual Studio/Visual Basic or Python/Tkinter/PAGE.

You are provided with the following requirements document:



REQUIREMENTS DOCUMENT

Date: January 10, 2019

Date Submitted:

Application Title: Burger Specials

Purpose: This Windows Classic Desktop application displays the burger specials for a restaurant named Farm Burger. The user can select one of two specials: Prime Beef or Veggie.

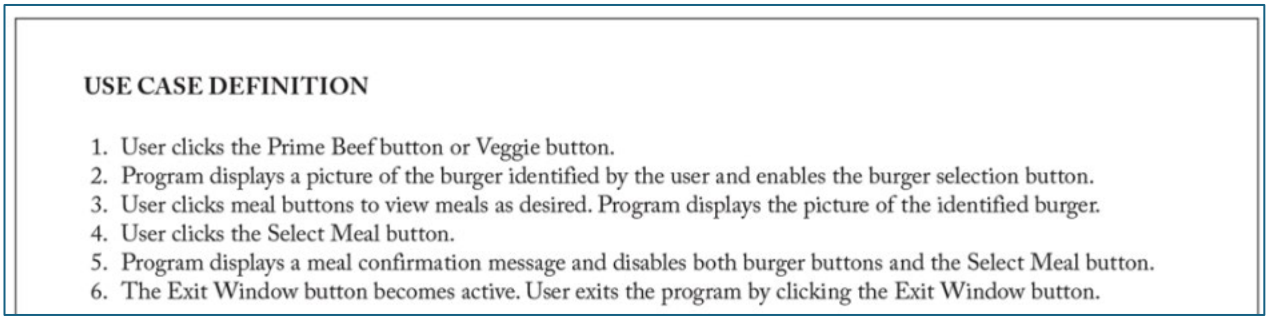
Program Procedures: From a window on the screen, the user orders one of two special burgers and the program confirms the order with a message.

Algorithms, Processing, and Conditions:

1. The user must be able to identify the Prime Beef or Veggie burger special name as many times as necessary until making a decision.
2. When the user identifies the special, a picture of that meal appears in the window.
3. Only one picture should be displayed at a time.
4. When the user selects Prime Beef, a message should be displayed to confirm the selection of the burger. In addition, the user should be prevented from identifying another meal after selecting a dish.
5. After the user selects a meal, the only allowable action is to exit the window.

Notes and Restrictions: A user can select a meal only after viewing one of the images of the meals.

And the following Use Case Definition:

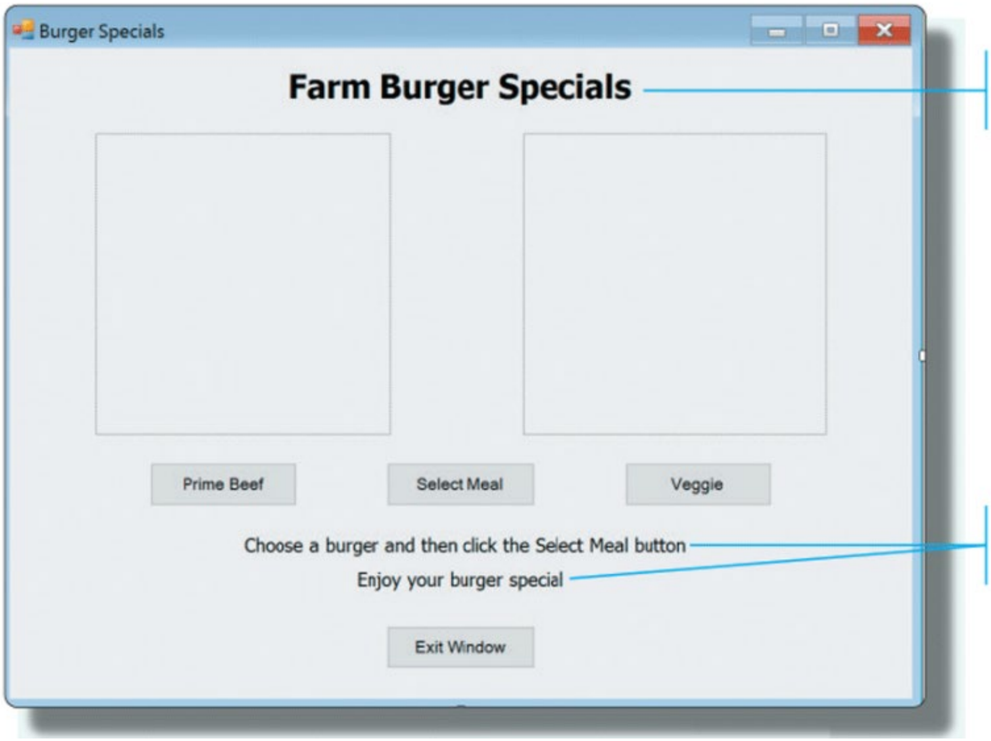


USE CASE DEFINITION

1. User clicks the Prime Beef button or Veggie button.
2. Program displays a picture of the burger identified by the user and enables the burger selection button.
3. User clicks meal buttons to view meals as desired. Program displays the picture of the identified burger.
4. User clicks the Select Meal button.
5. Program displays a meal confirmation message and disables both burger buttons and the Select Meal button.
6. The Exit Window button becomes active. User exits the program by clicking the Exit Window button.

A mockup of the UI is provided (see below) along with an initial description of the expected objects in the UI.

Screenshot of GUI



Farm Burger Specials

Choose a burger and then click the Select Meal button

Enjoy your burger special

Tahoma font, 16 points, bold

Tahoma font, 9 points

List of object properties

Type of Control	(Name) or ID	Text	Size	Font
The form	frmBurgers	Burger Specials	800, 590	
Label	lblHeading	Farm Burger Specials	350, 33	Tahoma 16 pt. **BOLD**
PictureBox	picPrime		260, 250	
PictureBox	picVeggie		260, 250	
Button	btnPrime	Prime Beef	128, 35	
Button	btnSelectMeal	Select Meal	128, 35	
Button	btnVeggie	Veggie	128, 35	
Label	lblInstructions	Choose a burger and then click the Select Meal button	360, 18	Tahoma 9 pt.
Label	lblConfirmation	Enjoy your burger special	172, 18	Tahoma 9 pt.
Button	btnExit	Exit Window	128, 35	

Assignment A3 - GUI Layout w/Event Plan

Deliverables include:

- an Event Plan (Word or Excel document) and
- a GUI Layout (created in either development environment)
- a project README Markdown file

Functionality is not required for the layout, only the creation of the GUI layout elements using the development tool. The Event Plan should include what will need to happen for each interaction with the interface per the requirements provided. Use the following template (or something like it) for your Event Plan:

Object	**Event Trigger**	**Event Processing**

Your deliverables (and grading rubric) are:

1. (5 points) A project README Markdown file with project name, student name, design tools used, and any issues encountered in Assignment A3. Also use the README to document particularly useful support sources (include URLs if applicable).
2. (20 points) A set of project files for VB or Tkinter that provide the GUI layout similar to the one presented above – code should be appropriately commented and any code taken from external sources should include URL references as citations. 10 points of the 20 for this deliverable will be based on code quality/readability; 10 points for closeness to the GUI mockup and elements.
3. (15 points) A Word or Excel document labeled A3EventPlan.docx or .xlsx (a PDF is also acceptable); the plan should include all the interaction elements and their expected triggers and processing behavior. Grading will be based on completeness of addressing objects and events.

These deliverables should be pushed to an accessible GitHub repository, and you should turn in a text file and/or comments to CougarVIEW with a link to the GitHub repo containing the above deliverables.

Assignment A4 – GUI Application

Implement the GUI specified by the documentation and design in Assignment A3. You may create the application in either of our current design environments. Image files required for use are contained in the assignment.

Your deliverables (and grading rubric) will be:

1. (5 points) A project README Markdown file with project name, student name, design tools used, and any issues encountered in Assignment A4. Also use the README to document particularly useful support sources (include URLs if applicable).
2. (35 points) VB or TKinter application with all project files required to execute the system – code should be appropriately commented and any code taken from external sources should include URL references as citations. Points breakdown:
    a. 15 points – All objects and events needed are represented in code and visually on GUI.
    b. 10 points – GUI should run as requested in development environment (VB or Python) – all events should occur as requested.

c. 10 points – Cleanly formatted code with comments. This should include a header block containing student, project, and class names. Comments for functions/methods/classes, comments for key actions or any statements that may not be obvious in their function. Use descriptive variable names.

These deliverables should be pushed to an accessible GitHub repository, and you should turn in a text file and/or comments to CougarVIEW with a link to the GitHub repo containing the above deliverables.

Note: Always cite what you write! If you get code or content from somewhere you must include at least a URL or other source identification. You must understand all the code you turn in. It is plagiarism (academic dishonesty) to use code or content, in part or in whole, written by other people without proper attribution. Failure to do so will result in a 0 on the assignment and may result in an academic misconduct report.

Example development lectures will be provided reviewing elements required for these projects.

See Bruce for questions.