PROG71985 – Winter 2019

Group Project

For the group project, you will work in teams – team sizes can be individual, 2 person or 3 person teams. Each different sized team will have a slightly different set of requirements to fulfil as part of the work for the project.

The project will revolve around the design of a "Bus Stop Announcement System" or BSAS. Do a Google search for this phrase to learn more about this concept. As noted on this web page:

http://www.ohrc.on.ca/en/next-stop-accessibility-report-public-transit-stop-announcements-ontario/appendix-b-stop-announcement-practices-and-commitments

the Region of Waterloo's Grand River Transit system was to have an automated system functional by 2010.

A BSAS has a relatively straightforward architecture:

- A GPS unit will provide time, latitude and longitude data once per second
- A database of bus stop data provides information as to support this process – details such as latitude and longitude of stops, bus stop names, route transfer data, and so forth will be present in this database
- A database of pre-recorded announcements that coincide with a specific geographical location – for example, if a bus is approaching and within 500m of a bus stop called "King and Weber", a voice message will be played to announce the pending stop, coupled with either an LCD video display or scrolling LED display that will provide a visual display of the upcoming bus stop to passengers. This supports accessibility requirements for all Ontarians.
- If a user has selected a bus stop, the announcement system must indicate that the bus will be stopping at the next available bus stop. This is done audibly as well as visually.
- At a stop, if equipped, the bus shelter should be able to display time to next arrival of a bus on that route. This helps the public determine how long they may need to wait for a vehicle to arrive. Audible announcements on pending arrival will be provided.

Assignment Requirements

The simplified BSAS you will implement will execute under Windows 7, 8 or 10. No other operating systems are required to be supported. The BSAS will be a combination of one or more applications that work together to solve the overall requirements.

The applications will include:

a) Implementation of a stand alone application that obtains latitude and longitude data from a GPS receiver (or reads from a data file once per second that provides a capture of latitude and longitude data points), and determines whether the current geographical position is within 500 metres of a bus stop. Bus stop data will be housed within a CSV formatted database file. The database will provide the following information:
    a. Bus stop number (an unique numerical identifier)

b. Latitude and longitude of the bus stop
c. Bus stop name
d. Transfer information (for example, a bus stop might connect to routes 3, 7, and 12). Assume only 12 routes maximum for this project.

The application will write results into a shared file. The shared file will contain the bus stop identifier if within 500m, or -1 to indicate no current bus stop in range. This file will be written to every second.

b) Implementation of an stand alone Win32 GUI application that will update a full screen window twice per second with information that includes:
   a. Current Date and Time (Day Name Month Day Year, HH:MM AM/PM – for example: Monday February 4, 2019 08:20 PM
   b. Upcoming bus stop name (or blank if no bus stop approaching)
   c. Bus route transfer information (if such data exists for the upcoming bus stop)
   d. Font size, colours for foreground text and background will be read from a CSV configuration file upon application start up.

   The Win32 GUI application will read a shared file to obtain bus stop identifier, and will use this information to retrieve the bus stop name and route transfer information via the bus stop database. The shared file will be written to by the BSAS utility software to provide updated information. Twice per second (500 ms timer), this shared file will be read to obtain the latest information in order to update the full screen window.

c) Implementation of a stand alone application that reads a bus stop identifier from the shared file, and if not -1, it will play a sound clip that provides an audible announcement that the bus is approaching a given bus stop. The announcement will include the name and bus route transfer information. Individual sound files will be implemented to store bus stop names, the audio for "Approaching stop", the audio for "Connections for", the audio for "route", and audio for the numbers from 1 to 12. The resulting audible output should sound as close as possible to a human reading a script such as "Approaching stop King and Weber. Connections for route 3 route 7 route 12". Check the shared file once every half second (when no bus stop is being approached), or immediately after completing audio output.

Deliverables

As teams can consist of single individuals, two students or three students (maximum), the deliverables for these various sized teams will be as follows:

- For single student team – choose one of (a), (b) or (c) for design and implementation
- For two student teams – choose two of (a), (b) or (c) for design and implementation
- For three student teams – implement all three of (a), (b) and (c) for design and implementation

At the due date, in the drop box provided, submit in a single ZIP archive with the following:

a) Source code and makefiles for each of the applications your team is required to implement. Code should be in OTBS.
b) Software Requirements Specification for each application developed by the team.
c) Software Design Document for each application developed by the team. Provide details on shared files, database formats, etc. that your team will use.
d) Database files, audio files and other data that supports your solution
e) Short report that documents basic details on how to build your application(s) and installation details in order for a 3rd party to install and execute your solution. In your report provide a reflection on how your solution can benefit society – your reflection can discuss, for example, impact for Ontarians with disabilities, as well as highlight any potential environmental benefits your solution may have in terms of making a bus system more efficient. Finally, estimate the cost of implementation and testing. Use an hourly figure of $50/hr for development time per person on the team.
f) A signed declaration for the team that documents which team member implemented which application.

Marking Scheme

This assignment will be graded out of 12 points per application submitted. For 2 or 3 student teams, the average of the grades for each submitted application will act as the team score out of 12, with the final 3 points coming from the final report and signed declaration. Grading is as follows:

Software Requirements Specification (SRS) – 2 points:

- 0 = no SRS provided
- 1 = SRS exists but provides minimal coverage of requirements of application(s)
- 2 = SRS exists and provides strong coverage of requirements of application(s)

Software Design Document (SDD) – 2 points:

- 0 = no SDD provided
- 1 = SDD exists but provides minimal coverage of the design of application(s)
- 2 = SDD exists and provides strong coverage of the design of application(s) including file formats

Use of tools - Multi-source file implementation – 2 points:

- 0 = solution not implemented as a multi-source file project
- 1 = solution has multiple source files, makefile provided but software does not build cleanly
- 2 = solution has multiple source files, makefile provided and software compiles cleanly

Professionalism in Software Engineering – 1 points:

- 0 = code has no discernable style
- 1 = OTBS styled code that is documented as expected (header comment blocks, function comment block, in-line comments explaining code, appropriate coding style used throughout)

Correct Implementation – maximum 5 marks deducted for coding errors

- Syntax errors and/or runtime logic errors will be reviewed for solution correctness

Total for above: 12 points

Final Report – 3 points:

- 0 = no final report or signed declaration provided
- 1 = final report / signed declaration exists but provides minimal coverage of build / install details
- 2 = final report / signed declaration exists and provides strong coverage of build / install details
- 3 = as per above, but report provides details on societal and environmental impact of solutions

Overall total: 15 points

Sprints

Supervised project time will be provided through three Sprint sessions. A recommended use of the three sprint sessions includes:

Sprint #1 (Week 7) – use this time to gather requirements, clarify functionality and begin SRS / SDD documentation effort – continue design work into Student Success Week

Sprint #2 (Week 10) – begin implementation – finalize design activities, research algorithms, write code, get assistance with debugging

Sprint #3 – (Week 13, 14) – final implementation, integration, testing

Project due April 19, 11:59PM

Appendix A – Graduate Attributes

Professional Body of Knowledge

KB	KB 1	KB 2	PA	PA 1	PA 2	PA 3	IV	IV 1	IV 2	IV 3	ED	ED 1	ED 2	ED 3	ED 4	ED 5	ET	ET 1	ET 2	ET 3	ET 4
Knowledge base	Facts	Concepts	Problem analysis	Decomposition	Methodology	Validation	Investigation	Research	Measure	Experiment	Design	Problem Def'n &	Preliminary Design	Detailed Design	Implementation	Verification & Valid'n	Use of engineering tools	Models/Simulations	Measurement Tools	Manufacturing Tools	CAD Systems
	D	D		D	D			D		D		D A	D A	D A	D A	D A				D A	

Employability Skills

TM	TM1	TM2	TM3	CM	CM1	CM2	PR	PR1	PR2	PR3	PR4	LL	LL1	LL2	LL3	LL4
Individual and team work	Personal Contribution	Collaboration	Infrastructure	Communication skills	Log Engineering Info	Convey Engineering Info	Professionalism	Work Ethic	Professional Conduct	Professional Contribution	Professional Practice	Life-long learning	Autonomous Learning	Applying Knowledge & Skills	Self Direction & Reflection	Learning Strategies (Metacognition)
	A	D	D			D A		D	D	D	D			D		

Professional Responsibility

SC	SC1	SC2	SC3	EE	EE1	PM	PM1	PM2	PM3	PM4	PM5
Impact on society and [illegible]	Environmental [illegible]	Product Life Cycle	Balance & Tradeoff	Ethics and equity	Ethical Responsibility	Economics and project management	Project Scheduling	Resource Allocation & Costing	Risk Management	Business Planning	Economic Analysis
	D	D			D		D			D	D

LEGEND		
I	**Introduced**	first experience/use
D	**Developed**	continued experience/use
A	**Applied**	integration/extension of knowledge & skills