Description of bug	Date	Root causes	Test type	Resolved by	How
After development of Scenario 2 was finished, the system would crash after trying to load Scenario 1	18/04	The Timeline class would attempt to create a background_loan instance, but there was no data for it in scenario 1.	Integration testing	Peter	Added checks in the form of 'if reference_loan != None' to every point where the Timeline class could interact with a loan. A new instance of the same bug was found on 25/04, which was fixed the same way.
System crashes after selling all owned stock. ZeroDivisionError in shell.	20/04	In Timeline_Stock class, the recalculation of total spent on stock includes a division of stock volume. If stock volume is reduced to zero after a sell, ZeroDivisionError occurs.	Validation testing (sellStock_3)	Lachlan	Division by volume is set to occur before volume is set to zero.
Stock price can decrease below zero	25/04	In Background_Stock, there is nothing to prevent stock price from dropping below zero.	Unit testing	Lachlan	Updated stock price is set to max(1, updated stock price). **Note:** this revealed a parallel bug in Background_Loan. Updated interest rate is set to max(0.2, updated interest rate) on 06/05.
Hovering over the information and graph button would not function after implementing scrollable panels	25/04	The scrollable panel entity in the Timeline class was layered on top of the buttons, so the panel hover function was called by the GUI manager instead of the button's hover function.	Regression testing	Peter	Changed the layering setting on the affected buttons so they would be on top of the panel, which would give them priority for hovering calls.
Graphs displaying historical interest rates (for loan) and historical stock prices (for stocks) weren''t scaling properly. - The y-axis limits were too close from the minimum and maximum values of the plot,	27/04	If not defined, limits and values on the y-axis are automatically set my Matplotlib	Unit testing	Mathilde	We manually set the y-axis limits (y_min and y_max) and restrict values on the y-axis to integers when creating the plot.

making the graph hard to read. - Values on the y-axis were displaying float numbers instead of integers for stock prices					
Montserrat font used in the Menu was not displaying correctly on a Windows computer	29/04	Montserrat font is present as a system font on Mac but not on Windows	Validation testing (non-functional requirement: portability. Refer to section 2.4 in the SRS)	Mathilde	We download the font file (montserrat_font.ttf) separately and import it when loading the font in the Menu
Moving onto the next level, UI elements of the previous level (timelines, buttons) appeared in the new level. User can interact with these UI elements	29/04	UI manager is global and UI elements are therefore not tied to the existence of the object that 'owns' them.	Integration testing	Lachlan	PyGame GUI UI manager class has reset_and_clear method which kills all of its UI elements. This method is called whenever user restarts a level or moves on to the next level.
html text wasn't displaying correctly in the "about", "win", "lose" and "tutorial" windows with unidentified symbols instead of lists and titles.	02/05	pygame_gui library does not recognize html symbols and (used for lists) nor <h2> and <h3> (used for headings)	Unit testing and Validation testing (non-functional requirement: portability. Refer to section 2.4 in the SRS)	Mathilde	We use interpunct symbols for lists and bold text for titles. We then realize interpunct is not recognized on Windows so we decide to use the universal "+" symbol
A scenario would end one time progress unit sooner than expected.	06/05	The stop condition in the Background class was set as 'the number of passed timesteps is equal to the total number of timesteps'.	Validation testing (progress_1)	Peter	The stop condition was set to 'the number of passed timesteps is larger than the total number of timesteps'.