BitExplorer
Bitcoin Transaction Visualizer

Jeff Lee
Lintao Ma
Narayan Timsina

December 11, 2017

Contents

Chapter 1

Daily Activities

1.1 November 6th

Laid groundwork, brainstormed ideas for project.

1.2 November 8th

Added Processing core to Java package which we will use for the visualization part of the project.

1.3 November 10th

Transaction Stream

- Gets stream of new bitcoin transactions through websocket.
- Parses JSON string to get relevant information.
- Saves from address, to address, and value.

1.4 November 13th

Balls and Movements

- created an animation of randomly bouncing Ball objects that can hold data

1.5 November 15th

Link Transaction and Balls

- Each new transaction is put in a Queue to be added.
- Queue is flushed and new balls are added to a list, appear in screen with random motion and size proportional to transaction value.

1.6 November 17th

Live Exchange Rate

- Live exchange rate is refreshed every 10 seconds from Blockchain API, displayed in screen.

Transaction Graph

- Graph groups of transactions in bar form, displays what size transactions are most popular

1.7 November 20th

Buttons

- Created standard button object to be used throughout application.
- Added on-screen buttons for easier user interaction (previously used keyboard).

GUI Upgrade

- Branded application, devised a logo.
- Made balls of certain values a set color to increase readability.
- Added a visually pleasing background.
- Made buttons responsive (change color on hover).
- title and footer bar to display important data.

1.8 Nomember 24th

Fetch Recent Block

- Gets the hash of the most recent block.

Block Object and Display

- Fetches JSON string given the most recent block.
- Parses through an average of 100,000 lines of data to find relevant info.

- Saves total transaction volume, biggest transaction, parent block hash, and more.
- Displays simple text on screen to show the information.

Blockchain Travelling

- Utilizes Blockchain linked-list style structure to allow the parent block or child block to be quickly loaded.
- Each block has a parent block address, parentBlock object, and childBlock object
- If the parent has not been loaded, we fetch the info
- If the parent has been loaded, it is saved in parentBlock.
- childBlock is set on creation, as every block must have a child.
- Latest Block is noted to user, does not have a child.

1.9 November 26th

Ball Data Display

- Wanted to display more data for each transaction ball.
- Now when stream is paused, every ball displays to and from address on hover.

GUI Upgrade

- Display block info in a clean frame.
- Easily switch from stream to explorer mode and back.
- Added startup screen to allow user to select a mode to boot into, but can switch back and forth later on.

Enlarged Window Size

- Made application window slightly bigger to allow for more features, less crowding.

Upgraded Graph

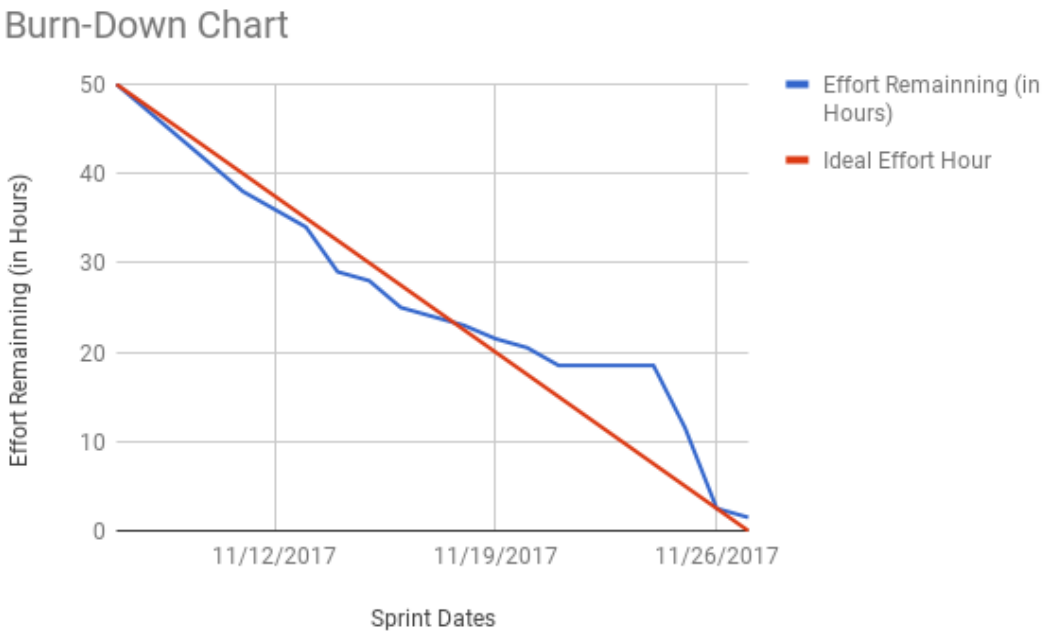
- Made four bars instead of three to allow for better value intervals.
- Set cap length at 500 to prevent overflow.

Color Key

- Added color key to show user what value each color ball represents.

Chapter 2

Burndown Chart



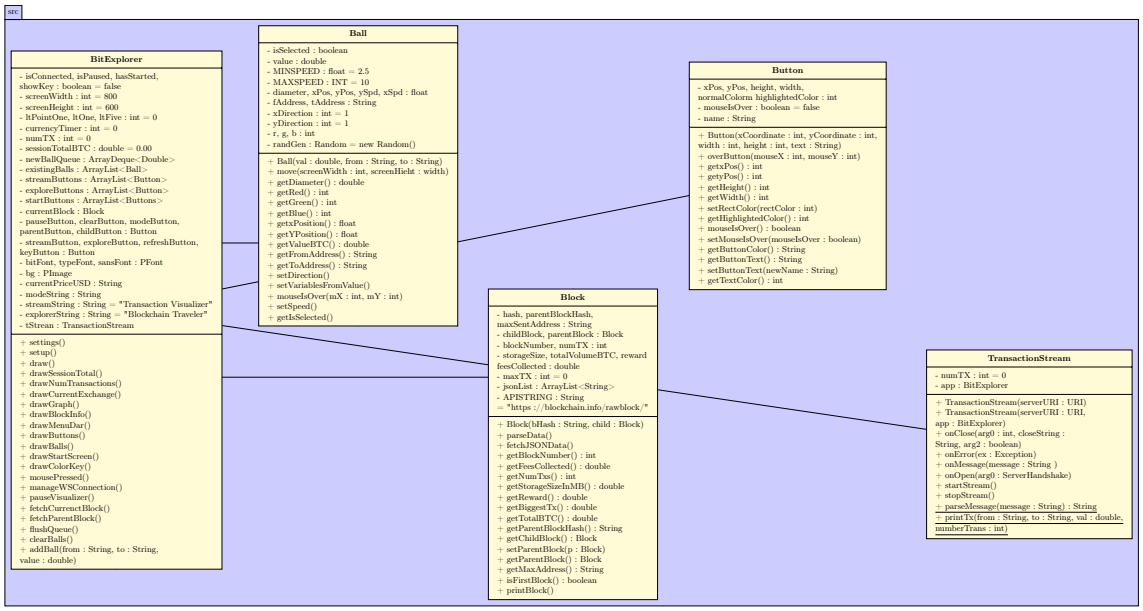
Chapter 3

Product Backlog

Story Name	Task No.	Task Description	Status	Estimated Effort (in Hours)	Effort Progress (in Hours)	Effort Remainning (in Hours)	Ideal Effort Hour	Sprint Dates	Sprint Dates
									11/15/2017
1. Transaction Stream	a.	Get straeam of new bitcoin transactions through websocket	Closed	2	0	50	50	11/7/2017	11/7/2017
	b.	Parses JSON string to get relevant information	Closed	3	0	47	47.5	11/8/2017	11/8/2017
	c.	Saves from address, to address, and value	Closed	1	0	44	45	11/9/2017	11/9/2017
2. Balls & Movement	a.	created an animation of randomly bouncing Ball objects that can hold data	Closed	4	0	41	42.5	11/10/2017	11/10/2017
3. Link Transaction and Balls	a.	Each new transaction is put in a Queue to be added	Closed	1	0	38	40	11/11/2017	11/11/2017
	b.	Queue is flushed and new balls are added to a list, appear in screen with random motion and size proportional to transaction value	Closed	1	0	36	37.5	11/12/2017	11/12/2017
4. Live Exchange Rate	a.	Live exchange rate is refreshed every 10 seconds from Blockchain API, displayed in screen	Closed	1	0	34	35	11/13/2017	11/13/2017
5. Transaction Graph	a.	Graph groups of transactions in bar form, displays what size transactions are most popular	Closed	2	0	29	32.5	11/14/2017	11/14/2017
6. Buttons	a.	Create standard button object to be used throughout application	Closed	2	0	28	30	11/15/2017	11/15/2017
	b.	Add on-screen buttons for easier user interaction (previously used keyboard)	Closed	2	0	25	27.5	11/16/2017	11/16/2017
7. GUI Upgrade	a.	Brand application, devised a logo	Closed	1	0	24	25	11/17/2017	11/17/2017
	b.	Make balls of certain values a set color to increase readability	Closed	1.5	0	23	22.5	11/18/2017	11/18/2017
	c.	Add a visually pleasing background	Closed	1	0	21.5	20	11/19/2017	11/19/2017
	d.	Make buttons responsive (change color on hover)	Closed	2	0	20.5	17.5	11/20/2017	11/20/2017
	e.	Title and footer bar to display important data	Closed	0.5	0	18.5	15	11/21/2017	11/21/2017
8. Fetch recent block	a.	Gets the hash of the most recent block	Closed	2	0	18.5	12.5	11/22/2017	11/22/2017
9. Block object and display	a.	Fetches JSON string given the most recent block	Closed	2	0	18.5	10	11/23/2017	11/23/2017
	b.	Parses through an average of 100,000 lines of data to find relevant info	Closed	2	0	18.5	7.5	11/24/2017	11/24/2017
	c.	Saves total transaction volume, biggest transaction, parent block hash, and more	Closed	2	0	11.5	5	11/25/2017	11/25/2017
	d.	Displays simple text on screen to show the information.	Closed	2	0	2.5	2.5	11/26/2017	11/26/2017
10. Blockchain Travelling	a.	Utilizes Blockchain's linked-list style structure to allow the parent block or child block to be quickly loaded	Closed	1	0	1.5	0	11/27/2017	11/27/2017
	b.	Each block has a parent block address, parentBlock object, and childBlock object	Closed	1	0			11/28/2017	11/28/2017
	c.	If the parent hasn't been loaded, we fetch the info	Closed	1	0			11/29/2017	11/29/2017
	d.	If the parent has been loaded, it is saved in parentBlock	Closed	1	0			11/30/2017	11/30/2017
	e.	childBlock is set on creation, as every block must have a child	Closed	1	0			12/1/2017	12/1/2017
	f.	Latest Block is noted to user, does not have a child	Closed	1	0			12/2/2017	12/2/2017
11. Ball Data Display	a.	Wanted to display more data for each transaction ball	Closed	2	0			12/3/2017	12/3/2017
	b.	Now when stream is paused, every ball displays to and from address on hover	Closed	1	0			12/4/2017	12/4/2017
12. GUI Upgrade	a.	Display block info in a clean frame	Closed	2	0			12/5/2017	12/5/2017
	b.	Easily switch from stream to explorer mode and back	Closed	2	0			12/6/2017	12/6/2017
	c.	Added startup screen to allow user to select a mode to boot into, but can switch back and forth later on	In-Progress	2	2			12/7/2017	12/7/2017

Chapter 4

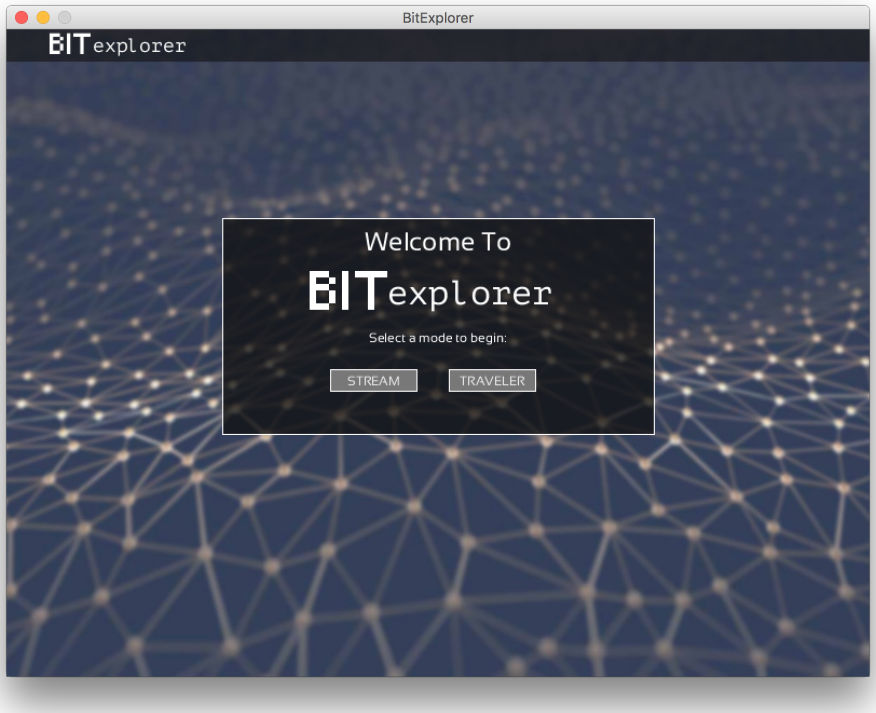
UML Diagram



Chapter 5

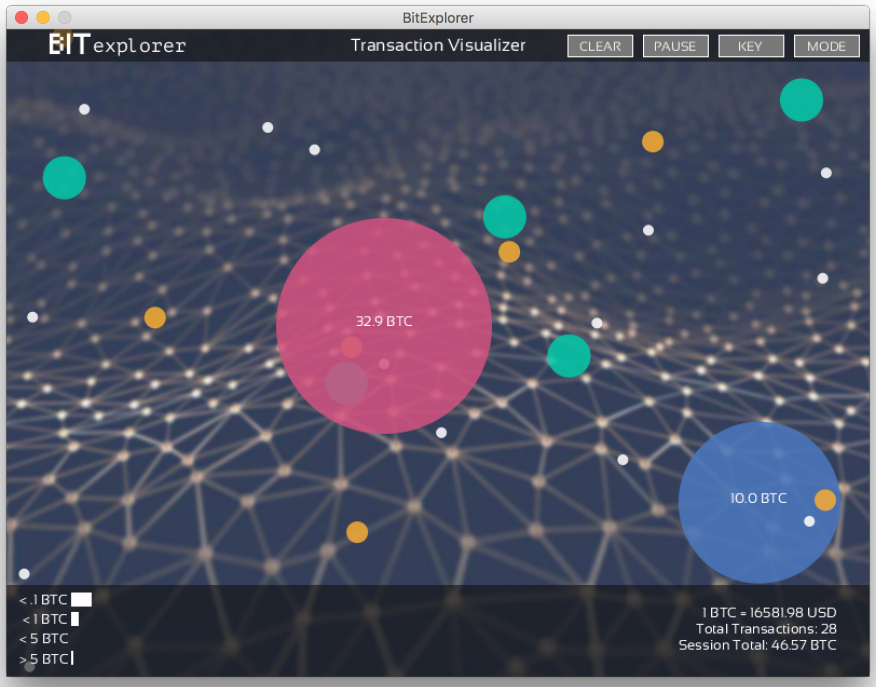
User Manual

5.1 Start Screen



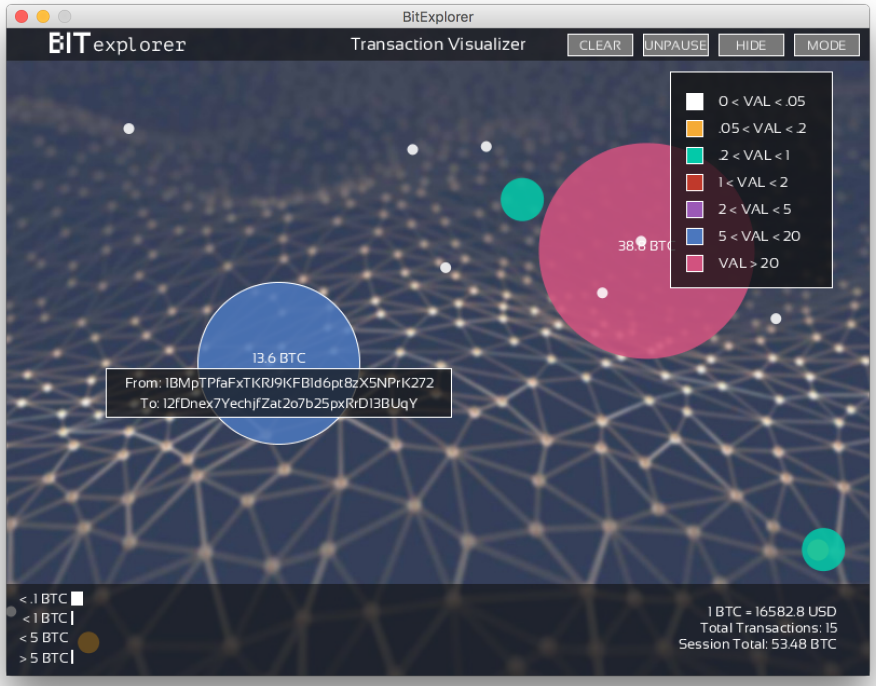
When BitExplorer is launched, a simple title screen appears that allows the user to boot into a specific mode: Explorer or Traveler.

5.2 Explorer Mode



If Explorer is selected, the frame is updated to show a live stream of all new incoming transactions. Each new ball is generated from the center with a random speed and direction, and a color and size correlated to its transaction value.

If the stream becomes to fast to process any information or if more details about a transaction are desired, the user can pause the visualization which halts any new transactions from being processed. While paused, a transaction ball can be hovered over to reveal the sender's and receiver's respective addresses. Additionally, a color key can be shown in the top right to easily see an approximate value of each transaction.



5.3 Traveler Mode

If the user boots into Traveler mode, the most recent block is fetched from the Blockchain API and the relevant information is displayed onscreen. Additionally, the user can use the mode button in the top right to toggle between modes after booting. The Traveler mode displays the most relevant information for each block, and can access each successive or previous block with the child and parent buttons below the display.