Team project - cover page

Kaizen Team

GUID 1

2

3

4

5

Contents

User roles

- User – includes player
- AI – automated player
- Robot – anything that is automated that is not a player including dealing cards

General priorities

- Must have – the playing of the game
- Should have – persistent data
- Could have – JavaScript GUI
- WLTH – all other fancy features

User stories

NO	Story	Conversations	Priority
1	As a user I want to be able to choose cmd mode.	Done initially via cmd flag. -c Basic reading of command line is in package.	MUST (anything that is cmd is a must)
2	As a user I want to open online mode.	Done initially via cmd flag. -o	COULD
3	As a user I want to load deck of cards.	There is only one deck of cards used, no need for a file chooser. This is by positioning 'StarCitizenDeck.txt' in home folder of the program. No specification to choose the file.	MUST
4	As a user(hereby U) – store persistent data on game.	Done using a store database calss in java.	SHOULD
5	As U i want to retrieve persistent data on past games.	Done using retrieve database class. Prompted at the beginning of the program.	SHOULD.
6	As U I want to receive the program state log in cmd mode.	Involves a number of attributes to be shown. Can be implemented with extra -t flag in cmd mode.	SHOULD
7	As U or AI I want to receive fair share of cards.	Implemented in main gameplay – faily easy for 2, 4,5 players. In the	MUST

		case of 3 players we need special rule.	
8	As a robot I want to randomize cards.	Done through array list collections shuffle method.	MUST
9	As a user I want to see details of top card.	Implemented by wrapping detail in card object, and card object in array list. Easy to retrieve.	MUST
10	As AI I want to choose best option for top card.	Choose an arraylist of details and choose max. Could implement hash set here.	MUST
11	As a robot I want to choose random first player.	Get random int from 1 to x, where x is the max number.	MUST
12	As a robot I want to generate relevant sql statements for the user for both input and output to the database.	Java class for postgres implemented. Each of the relevant input output info should be implemented in a variable in the classes.	SHOULD
13	As U I want to choose between online and cmd mode.	Easily done with a Boolean variable and cmd flags.	COULD
14	As U I want to see round, active player and card drawn (other details as well if needed).	Printed in standard out and in some way in html for web development.	MUST
15	As U I want to select category for the round.	Choose a number for specific category.	MUST
16	As a robot I want to determine who wins or loses the round.	Count cards is probably most reliable way but there should be other ways to do it.	MUST
17			
18			

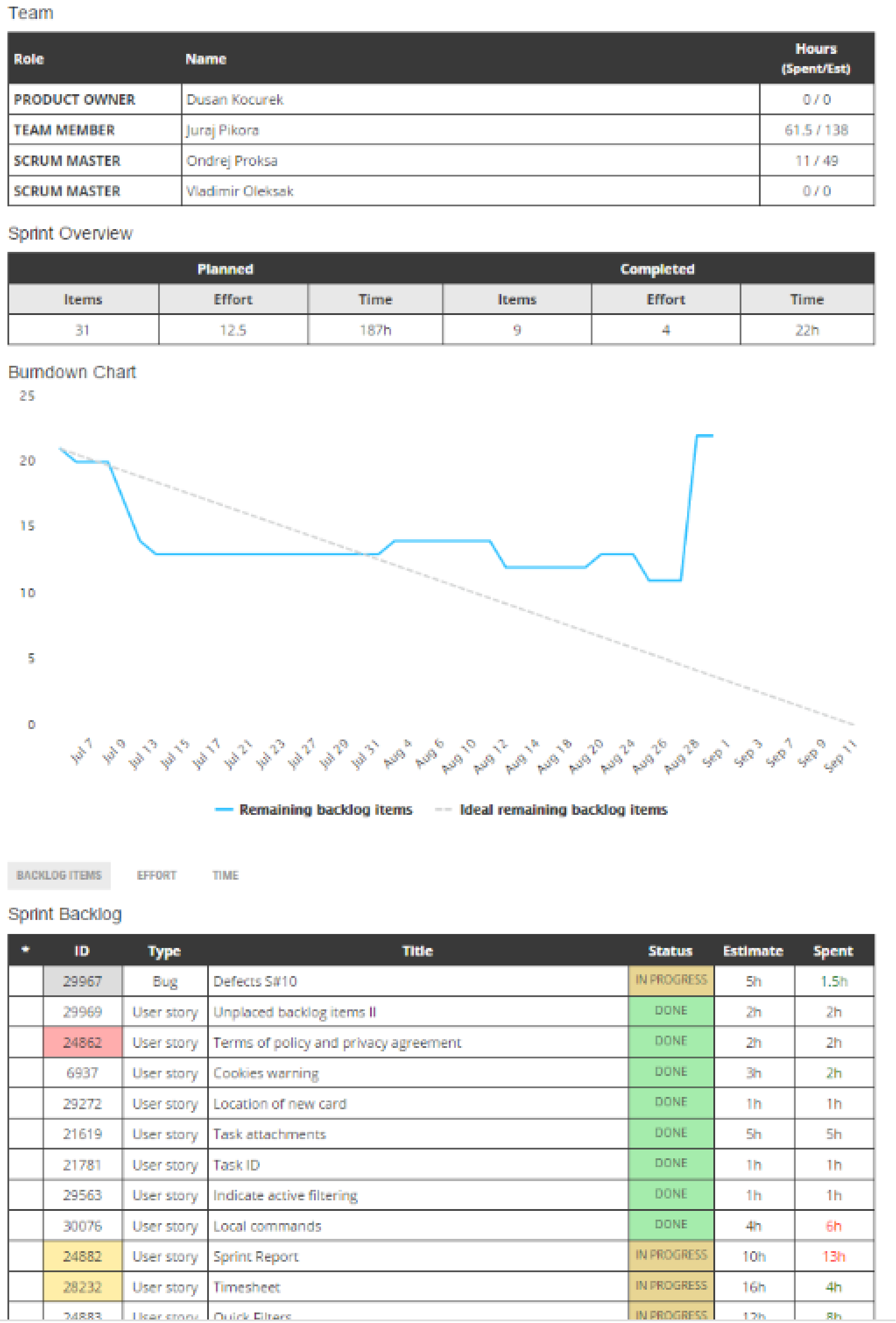
X	As a user I would like to choose from gui or command line the relevant deck .	Can be implemented in command line as args[x] or using JFileChooser.	WLTH
Y	As a user I want to see card artwork in GUI.	Drawn and added as image or png on web.	WLTH

Story points,estimated and actual time to be determined.

Non functional requirements:

- Postgres sql database use
- Cli used as per template package;
- Implement maven as per template package;
- Java for main application, javascript for web development.
- Agile approach – scrum.
- Jar file method used for running.
- …

Sprint 1 (also for 2) review and report:



Team

Role	Name	Hours (Spent/Est)
PRODUCT OWNER	Dusan Kocurek	0 / 0
TEAM MEMBER	Juraj Pikora	61.5 / 138
SCRUM MASTER	Ondrej Proksa	11 / 49
SCRUM MASTER	Vladimir Oleksak	0 / 0

Sprint Overview

Planned			Completed		
Items	Effort	Time	Items	Effort	Time
31	12.5	187h	9	4	22h

Burndown Chart

Sprint Backlog

*	ID	Type	Title	Status	Estimate	Spent
	29967	Bug	Defects S#10	IN PROGRESS	5h	1.5h
	29969	User story	Unplaced backlog items II	DONE	2h	2h
	24862	User story	Terms of policy and privacy agreement	DONE	2h	2h
	6937	User story	Cookies warning	DONE	3h	2h
	29272	User story	Location of new card	DONE	1h	1h
	21619	User story	Task attachments	DONE	5h	5h
	21781	User story	Task ID	DONE	1h	1h
	29563	User story	Indicate active filtering	DONE	1h	1h
	30076	User story	Local commands	DONE	4h	6h
	24882	User story	Sprint Report	IN PROGRESS	10h	13h
	28232	User story	Timesheet	IN PROGRESS	16h	4h
	24883	User story	Quick Filters	IN PROGRESS	12h	8h

Blablabla some comments on how it went must have planned and actual velocity.

Gantt chart.

Burndown chart

How much is left as seen in diagram above.

Assumptions

- compile from document

- get some of your own

Test cases

- test per story card

- Junit may be redundant in this case

Deficiencies

- will write at the end;

Screenshots:

- will be done towards the end;

Architecture points:

- model = this is storing data essentially from the relevant cards

- view = this is the gui or command line

- controller = anything that has to do with the non-static nature of cards

- player = object

- card = object

- everything else is implemented as we go – cards themselves may change