DM TOOLS

Mid-semester Reviews

SEP Team 5

Product Overview (Review)

Goblin

Small humanoid (goblinoid), neutral evil

Armor Class 15 (Leather Armor, Shield)
Hit Points 7 (2d6)
Speed 30 ft.

STR	DEX	CON	INT	WIS	CHA
8 (-1)	14 (+2)	10 (+0)	10 (+0)	8 (-1)	8 (-1)

Skills Stealth +6
Senses Darkvision 60 ft., Passive Perception 9
Languages Common, Goblin
Challenge 1/4 (50 XP)

Nimble Escape. The goblin can take the Disengage or Hide action as a bonus action on each of its turns.

Actions

Scimitar. *Melee Weapon Attack:* +4 to hit, reach 5 ft., one target. *Hit:* 5 (1d6 + 2) slashing damage.

Shortbow. *Ranged Weapon Attack:* +4 to hit, range 80/320 ft., one target. *Hit:* 5 (1d6 + 2) piercing damage.

Acolyte

Medium humanoid (any race), any alignment

Armor Class 10
Hit Points 9 (2d8)
Speed 30 ft.

STR	DEX	CON	INT	WIS	CHA
10 (+0)	10 (+0)	10 (+0)	10 (+0)	14 (+2)	11 (+0)

Skills Medicine +4, Religion +2
Senses Passive Perception 12
Languages Any one language (usually Common)
Challenge 1/4 (50 XP)

Spellcasting. The acolyte is a 1st-level spellcaster. Its spellcasting ability is Wisdom (spell save DC 12, +4 to hit with spell attacks). The acolyte has following cleric spells prepared:

Cantrips (at will): *light, sacred flame, thaumaturgy*

1st level (3 slots): *bless, cure wounds, sanctuary*

Actions

Club. *Melee Weapon Attack:* +2 to hit, reach 5 ft., one target. *Hit:* 2 (1d4) bludgeoning damage.

Things the narrator keeps track of:
- Where are the characters?
- How healthy are the characters?
- What *can/would* each of the characters do?
- *How* can each of the characters do this?
- What does the character have with them and why?

And more...

<u>Our Goal:</u> Providing a centralized means of organizing and generating details for playing through encounters inside a campaign

Framework Plans

Sprint 1:
- Electron and Electron Webpack
- Material UI
- Enzyme and Jest Testing
- TypeScript
- PostgresSQL
- Azure Hosting
- React
- Node JS
- Docker

Sprint 4+:
- Electron and Electron Webpack
- ~~Material UI~~ ***Bloomer/Bulma***
- ***TS***-Enzyme and ***TS***-Jest Testing
- TypeScript
- PostgresSQL ***and TypeORM***
- ~~Azure Hosting~~ ***Amazon Web Services***
- React ***and React Dev Tools***
- Node JS ***with Hapi, Joi, and Boom***
- ***Nock (Http Mocking)***
- ***React Router***
- ***Electron Store (Cookie Management)***
- ***JSON Web Token (JWT)***
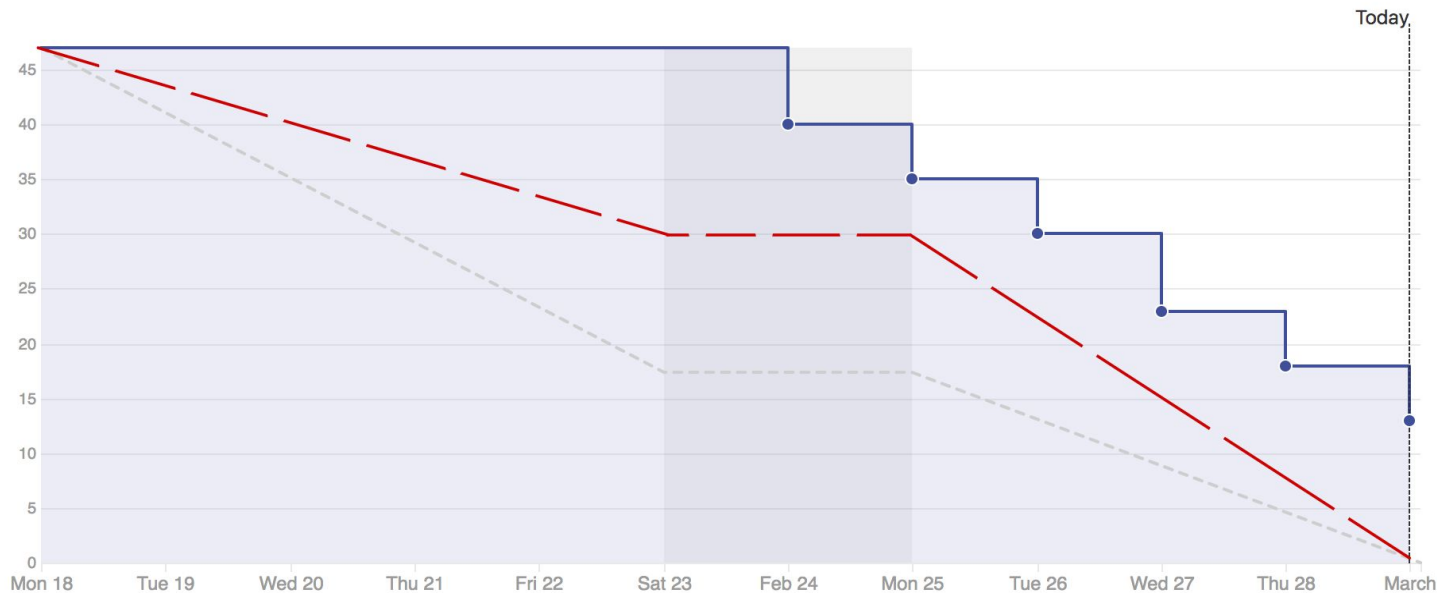- Docker

Sprint 1 Milestones

- Basic application setup
- Completed database schema
- Frontend and Backend for login and registration completed
- Frontend for monster creation started
- Initial lofi sketches made
- Adding Electron, Node, Docker, AWS, TypeORM, and Material UI

- 91% code coverage
- 45 point capacity
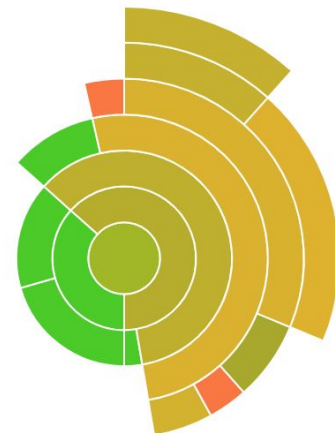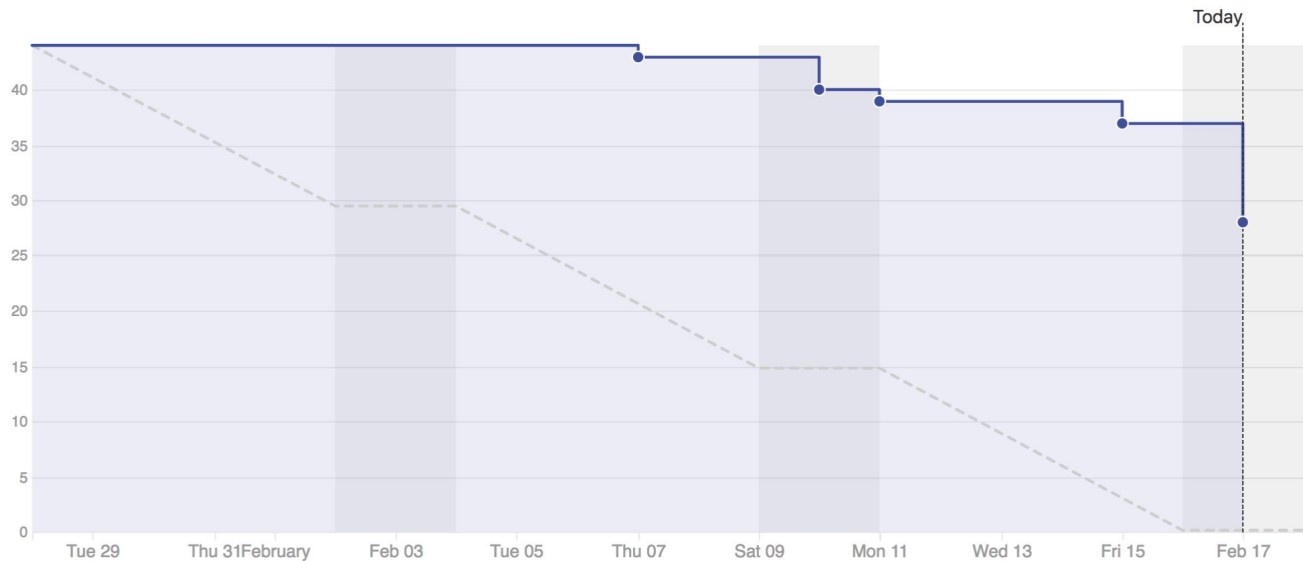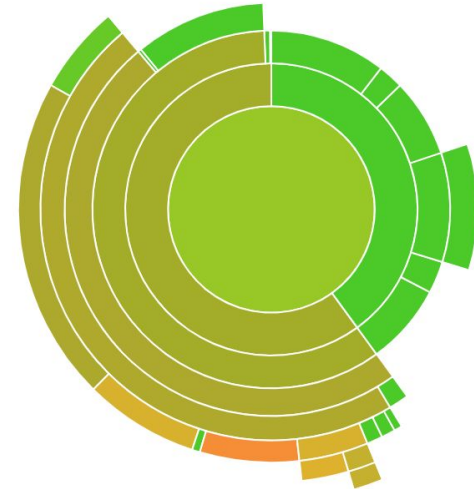
Sprint 2 Milestones

- React Router added (+ navbar)
- Session tokens and cookie management started
- Database populated with pre-gathered data for current features
- Prior features integrated from sprint 1
- Switch to TS testing frameworks, add devtools, add type checking to backend, and http mocking

- 92% code coverage
- 43.8 point capacity

Sprint 3 Milestones

- Material UI -> Bulma and Bloomer
- Secured application via session token checking for pages
- Cookie management
- Optimized API URL referencing and Docker Building
- Updated DB w skill data
- Create encounter

- 93% code coverage
- 52 point capacity

Future Milestones

MVP:
- Update and Delete Monster **[Sprint 4]**
- Read, Update, Delete Encounters **[Sprint 4]**
- CRUD Campaigns **[Sprint 4]**
- Paginated Views of Catalog Entries (Monsters, Encounters, Campaigns) **[Sprint 4 - Sprint 5]**
- CRUD basic entries for player characters **[Sprint 5]**
- Running through an Encounter **[Sprint 5 - Sprint 6]**

- -

Enhancements to MVP:
- CRUD for additional catalog entries (Items, Locations, etc) **[Sprint 6]**
- Enhancements to running through an encounter (Auto generated fields, etc) **[Sprint 6]**

Product Demonstration

Questions? Comments? Concerns?