Kenrick Rilee - *kenrick.rilee@gmail.com* - *cricklet.github.io* (full resume & projects)

ACADEMICS:

Princeton University: (originally 2013, now 2015)

- BSE, Computer Science.
- Took a leave of absence between 2012-2014 to co-found a startup and work in industry.
- Relevant coursework: advanced graphics, networks, computer vision, operating systems, algorithms & data structures, systems, computational physics, number theory.

Eleanor Roosevelt High School Greenbelt, MD (2005-2009)

- Valedictorian: class rank 1st out of ~800 students.

WORK:

2014: DevOps intern at Venmo

- Used puppet to setup stack monitoring and analytics.

2013-2014: Engineering fellow at Codecademy

- Full-stack (ruby/rails/js/Backbone) web development.
- Owned the ideation, prototyping, and implementation of some live-coding widgets.

2012-2013: Co-founded Rex/Mapsuaurs (defunct)

- TigerLabs University Accelerator (summer 2012), raised a small angel round (late 2012).
- As one of two main co-founders, my responsibilities (in addition to coding) included general project management and investor/partner relations.
- We built an Android app recommendation service that was downloaded by 60,000 users.

2011: Dead Arcade

- Led a team of two other people. Built an Android game that garnered 20,000 downloads.

AWARDS:

- 2nd place, Princeton Facebook Hackathon (2011): built a RTS game from scratch in 22 hours.
- 2nd place, Hack Princeton (2012): built an Android app recommendation algorithm.

PROGRAMMING:

Android + *Java*:

- *Rex*: app exploration/recommendations, search engine, batch uninstaller. (*60,000 downloads*)
- *PanesLibrary*: open-source library for creating flexible phone/tablet apps. (*300 stars on GitHub*)
- *AutoWallpaper*: updates your wallpaper with images scraped from Reddit's API. (*10,000 downloads*)
- *Dead Arcade*: 2D platformer, built completely from scratch. (*20,000 downloads*)

C or *C/C++*:

- Built an OpenGL renderer using modern C++, implementing VSM, SSAO, and hatching.
- Built an OpenGL top-down shooter, implementing boid AI, billiard ball physics, lightmaps, etc.
- Built an AI that uses particle filters, object detection, and range finding to play Wolfenstein 3D.
- *Advanced Graphics* (A): path tracer, laplacian mesh editing, image analogies.
- *Graphics* (A): ray tracer, mesh manipulation, shaders, openGL, etc.
- *OS* (B): boot-loader, kernel, scheduling, virtual memory, file system.
- *Networks* (A-): simple server, HTTP proxy, router, simple TCP.

JavaScript and *HTML5*:

- *Codecademy*: prototyping features, feature development, refactoring, chai testing.
- *Rex frontend*: prototyped versions using canvas, SVG, and D3.js.
- Built a game scratch using <canvas/> for the LD 48 hour game-jam. (*rated top 6% for graphics*)

Python:

- *Rex backend*: crawler, recommendation engine, search algorithms, events tracking, and analytics.

Other: Unity3D, Ruby/Rails, Backbone, Mongo, Redis, OpenGL, GLSL, Puppet, Haskell, Flask