

KEVIN TURNER

Email: kevin@kpturner.co.uk Tel: 07834 867862 LinkedIn: <https://www.linkedin.com/in/kevin-turner-56551>

Summary

Until parting company with CoralTree Systems Ltd (of which I was a co-founder in 2004) after the company was sold, I worked as the Technical Director responsible for all software development and R&D. A well-known contributor to many international online software development communities, most notably within the IBMi sector, it was my responsibility to ensure that CoralTree and its products remained up-to-date with industry standards. I am a tenacious and fairly obsessive individual who likes to make the very best of my abilities, whether it be developing software for the latest technology or competing in my favourite sport. Having spent more than 30 years working with enterprise-level applications, I have a full understanding of the fast-paced nature of the industry and a wealth of experience in all aspects of software development. Despite my seniority in the CoralTree business I am a hands-on full stack application developer both within a work environment and (sadly) in my spare time. The last four years being almost exclusively working with node.js on Unix, Windows and IBMi both with CoralTree and my other company, K P Turner Ltd. I am also a Director of a property management company called Lawturn Major Ltd.

Skills

| Languages | Libraries/Frameworks | IDE | Databases | Change control | Other tools |
|------------------------|----------------------|---------------|------------|----------------|----------------------|
| Node.js/Typescript/NPM | jQuery | VS Code | DB2 | Git | AWS |
| JavaScript | AngularJS | Visual Studio | MySql | SVN | Terraform |
| RPGLE | Typescript | Eclipse | PostgreSQL | SEE/Change | Ansible |
| CLLE | Bootstrap | | SQL Server | Turnover | Jenkins |
| HTTP/Web sockets | Require.js | | Oracle | | Travis |
| HTML/CSS/JSON/XML | Sails.js | | Redis | | Mocha |
| VB.Net | Express.js | | MongoDB | | Cucumber |
| C# | Vue.js | | MariaDB | | Gherkin |
| Java | Nuxt.js | | | | Heroku |
| PHP (not expert) | Waterline | | | | BugZilla |
| C (not expert) | Async.js & Promises | | | | Jira |
| Synon/2E (Cool/2E) | CGIDEV2 | | | | jMeter |
| Adelia | Apache | | | | Webperformance |
| REST | nginx | | | | Exposure to Selenium |
| SOAP | Lightstreamer | | | | |
| | Grunt | | | | |
| | SASS | | | | |
| | Waterline | | | | |

Work history

Elsevier – Contract Software Developer

12/2017 to present

Publishing industry. Contract node.js back-end developer in a DevOps environment. Multiple applications deployed on AWS, using GitHub, Travis and Jenkins. Infrastructure managed with Ansible and Terraform. TDD/BDD testing with Mocha and Cucumber. Scrum based - responsible for numerous enhancements and bug fixes working closely with small devops team. Some smaller projects using Vue.js and Nuxt.js.

K P Turner Ltd - Director

09/2017 to 11/2017

Although only incorporated in Sept 2017 after parting company with CoralTree, I have been doing a number of things externally under this guise for several years:

Node.js/Angular: Events Management

Full stack MEAN application using **MySql** (instead of MongoDB), **Express** (sails), **AngularJS**, **Bootstrap** and **node.js**. Deployed on **Heroku** and my own Centos server. Developed for the Masonic Province of Hampshire and Isle of Wight to

manage events/bookings. The UI is designed for be easily usable for all ages (people in their twenties to people in their nineties) and therefore functional as opposed to stunningly beautiful and full of eye-catching graphics. It works on a smartphone, tablet or desktop (responsive). The node.js back-end uses Express (actually the Sails.js wrapper) makes full use of csrf for security. Authentication is handled using passportjs (<http://www.passportjs.org/docs>) which exposes Social Media authentication via Facebook, twitter, Google+ etc or direct APIs like OAuth, Oauth 2.0, Bearer, or simple user name and password combos. The GitHub repository is here: https://github.com/kpturner/sails_events
The application can be seen here: <http://squareevents.org>

A test version: <http://test.squareevents.org> Please be aware that registering to use this application will only show you the public facing features, not the full administrative features (although this can be arranged).

The application is also used by the Province of Surrey, The Masonic Trout and Salmon Fishing Charity and the H&IoW Provincial Grand Stewards Lodge.

Currently undergoing an Angular 5 rewrite.

Node.js/Sails/Express: *Chromis*

GitHub (<https://github.com/kpturner/chromis>). Chromis is a back-end application framework designed to transact JSON data model(s) with any client (e.g. Angular 1,2, 4 or 5) or the CoralTree proprietary Renaissance UI engine using HTTP (XmlHttpRequests) and/or web sockets. Chromis is completely platform agnostic and is loosely based on **sail.js** which in turn is a wrapper for **express.js**. Chromis is designed to serve multiple database instances and uses an ORM called **Waterline** to seamlessly enable the Framework to switch between (for example, PostgreSQL, MySql, MongoDB – in fact any database supported by Waterline). For scalability the framework will start and manage back-end servers to handle requests and communicates via a **redis** based message bus called busmq (<https://github.com/capriza/node-busmq>). Session management is handled either via redis or JWT – the latter being the preferred default.

Node.js/socket.io: *Web socket server*

A node.js application/server designed to provide web-socket services to any client – similar to commercial products like Lightstreamer but much more light-weight. This implementation is predominantly designed for the IBMi platform and expects “events” to be enqueued for broadcast on data queues by legacy back-end applications. It is in use in conjunction with a CoralTree Renaissance application in Germany serving over 5000 concurrent call centre agents.

GitHub: https://github.com/kpturner/kpt_ws

Consultant to Zentive Ltd

Zentive have me on a retainer to provide emergency support (on a best endeavour basis) for all aspects of their IBMi server should I be needed.

Other

Some smaller projects using **React**, but as these are not yet commercially used applications I have not listed it as a key skill.

CoralTree Systems Ltd - Technical Director and Co-Founder

02/2004 to 09/2017

Lead technical architect (hands-on developer). We grew the company from 7 in 2004 to 60+ development, QA and support staff by 2017, and I was the lead architect/consultant responsible for all aspects of our software development. CoralTree develop subscriber management, billing and CRM solutions in the Cable, Satellite, Telephony and Broadband industry in the UK, Germany, CEE and SA. CoralTree applications are all based on an in-house full stack browser based SPA framework called Renaissance which can either utilise a platform agnostic **node.js** back-end and API suite (called Chromis) or legacy green screen RPGLE/AS400/IBMi systems via web services and APIs. In Germany alone CoralTree’s CRM supports 5000+ agents, all working concurrently via their browser. The framework incorporates JavaScript (jQuery, jQuery UI, backbone.js, require.js), HTML5, CSS, node.js, Web Sockets, Java, SQL. The IBMi/RPGLE version of the Framework was made available as an open source project under the MIT licence: <http://www.renaissanceframework.com/>

From a QA perspective CoralTree utilised automated tools like Selenium although I did not use it in anger myself (this area of testing was undertaken by a dedicated QA team). From a development perspective we had started to migrate from a waterfall approach to Agile/Scrum where customers were open to the change of approach – as a software vendor it was not easy to convince all customers that this would be a better way of working.

Renaissance UI

CoralTree proprietary framework similar to Angular 2. We created our own mark-up language called RML, which is a hybrid of XML and HTML designed to enable a UI to be defined by grouping/nesting Renaissance Components together. These components can be anything from a simple piece of text to a fully functional datagrid, and are written using the jQuery UI widget pattern. Each function exchanges a data model (JSON) with the back end either via XHR or websockets. The data model is immutable and is maintained in a similar way as that proposed by **Redux**. Renaissance applications are fully responsive and utilises a 24 column grid system which is similar to **Bootstrap** (which only allows 12 columns). Some background information here: http://www.coraltreesystems.com/wiki/index.php?title=Developing_in_Renaissance_6

VB.Net/C#: Equibook Equestrian Centre Management

Browser-based application written in VB.Net (and some C#) to provide a complete equestrian centre management function. MySql database, running on Windows Server platform in IIS. I wrote 90% of the system before engaging two other developers for maintenance, enhancements and support.

VB.Net: SMS Gateway

Windows desktop application provides a bi-directional SMS Gateway between any SMSC provider (i.e Materna or Vodafone) and any back-end system that can communicate with the Gateway via HTTP or sockets or web services. Supports SMPP, UCP, CIMD/CIMD2 and SEMA protocols, uses Windows Services and a feature rich MDI UI. Written 100% by me and used by Unitymedia in Germany for broadcasting information to engineers and customers and for inbound service requests and orders.

IBMi/AS400

With over 30 years' experience on this platform, I was the main point of contact for all CoralTree IBMi customers for consultancy, troubleshooting, 2nd level support

Convergys EMEA - Software Development Manager

1990 to 2004

Software development manager running large scale development teams/projects on the IBMi in the Cable/Satellite, Telephony and Broadband sector specialising in the AS/400 and development languages thereon. Formerly Swift Management Services. Hands on RPGLE, Synon2/E developer as well as managing a large development team covering both RPGLE, Synon and Java projects.

John Wiley & Sons – Project Leader

1988 to 1990

Project leader on the AS/400 using RPG and a well-known case tool called Synon2/E (latterly Cool/2E). Publishing industry.

Pacific Associates – Project Leader

1986 to 1988

Project leader on the AS/400 using RPG and a well-known case tool called Synon2/E (latterly Cool/2E). Software vendor and IBM business partner

Chase Manhattan Bank – Project Leader

1985 to 1986

RPG Project leader on System 38 – Banking industry

John Wiley & Sons – Senior Analyst Programmer

1984 to 1985

RPG Analyst programmer on the System 38. Publishing industry

Skandia Life – Analyst Programmer

1982 to 1984

Analyst programmer on the System 38. Life Assurance and Pensions industry

