

Brian Bennett

Email: bbenntgames@gmail.com

Website: www.bbennettgames.com

Objective

To apply relevant knowledge as a Software Engineer to program and support software solutions for problems in the video game industry involving but not limited to server architecture, development release pipeline, security, UI development, database management, feature implementation, and game design.

Summary of Experience

6 Full years experience in the Games Industry

Shipped Titles

Madden 17, Madden Mobile Seasons 4-9, NBA Live Mobile

Work History

2019-Current Electronic Arts

Job Title: Software Engineer II

Position Details: All previous responsibilities, but now in a more senior role on the team. Taking additional responsibilities into account when new efforts need to be put forward specifically in the areas of client and server architecture, tools, feature development and load test.

- Implemented front-end animations for user's superstar. This includes attainable celebrations, dances, defeat animations, level up animations and welcome animations.
- Added animation flagging system to this version of Ignite to allow us to flag parts on animations and send those flags to the rest of the game as they occur. This allowed all other systems from UI to game play to know when animations reach certain points.
- Created front and back-end for content driven purchasable and unlockable superstar animations.
- Created content driven system to add binding driven animation layers to the superstar element. This allowed content to define their own animations that could play along side the superstar animations.
- Updated blitz and elimination tournament leaderboard initial placement to be OVR based. Added leaderboard inactivity decay. Assist in leaderboard pooling creation.
- Led international team to update Madden Mobile to use Firebase ads implementation and AB testing.
- Updated import tool to allow us to import player data and convert it for the new season. Ran import and ensured all active users were brought over.
- Implemented front and back-end trades feature for Madden Mobile. Trades allowed users to trade cards and currencies while also having to complete conditions or achievements to complete for rewards. Added functionality for partial requirement progress to complete trade.

- Designed and implemented guts feature allowing artist to only create individual pieces for various elements in our game like cards or rewards and allowed content to put the pieces together as they see fit. This moved time off our artist which was a bottleneck for our content creation.
- Design architecture for calendar reward feature.
- Created look up system back end. This allowed users to easily search for where to acquire certain items in the game. Allowed for content driven overrides so we could point players to certain activities over others.
- Implemented multiple layers for our map feature and added parallax effect on these layers. The layers would move at a percentage of the size of the map.
- Included score/stat-based conditions to allow content to define events that are not time based.
- Assisted in marketplace implementation, an internal tool that allows multiple game servers share user data. In Madden Mobile's case, we used it to share a Madden level, certain currencies, animations and superstar equipment across mobile and console version of Madden for multiple years.
- Assisted in architecting and implementing OVR banding feature. This allows content to define OVR bands and move them over time. Starter cards would all start at the bottom of the band and be updated when the band moves.

2017-2019

Electronic Arts

Job Title: Software Engineer I

Position Details: Full-time member of the Madden Mobile team within EA, working more specifically on features, tools, and supportive development work for Madden Mobile.

- Assisted in card pack odds feature. Calculated odds for each card the pack can give. Created content driven system to only show certain odds for card groups. Implemented front-end for displaying card pack odds.
- Assisted in implementation of synchronous PvP mode, Overdrive. Implemented match state synchronization for both player match states. Assisted in tactics feature activation for both match states.
- Implemented front and back-end for store features. Implemented android and IOS SKU purchasing and receipt validation. Created front-end UI for store along with multi-layer categories of items. Implemented limited time offer pop up system.
- Created front-end UI for league survival feature. League members take part in events getting progressively harder to survive as long as they can for rewards. Ensured all league members had updated information on all league members.
- Assisted in creation of weekend tournaments, blitz tournaments and elimination tournaments PvP modes. Implemented various parts of front and back-end for all these features.
- Implemented deep links for IOS and Android. Set up links to navigate to certain parts of the game, start friendly matches with friends or accept friend request. Encrypted game specific data in deep link when user generated to hide user data.
- Implemented AB testing tech using Swrve. Implemented back-end REST API to send user data to Swrve server and fetch AB testing data.

- Updated leaderboard tech to support multiple leaderboards. Updated Redis cache and SQL data base to support this as well. Implemented jobs to updated data base with Redis cache. Updated front-end for specific features as well.
- Ensure that dev, stage, and prod java server/client environments are in a good state and respond quickly and efficiently when they are experiencing issues to return them to a good state.
- Updated and maintained load test for each release. Ensured peak simultaneous users could be maintained with every additional feature or existing feature change. Ensured other test results passed as well such as rolling deploy, high ramp rate, bad app-server state and client error state under load.
- Penetration tests that include memory modification, RPC spoofing, and various other attacks on game servers/clients in reaction to bad actor activities as well as preventative exercises.
- Optimization of Content Creators, Artists, and even Engineer workflows for all parts of the team.
- Created new achievements and user conditions tracking various user data throughout the game. Assisted in creation of user condition system.
- Address bugs in a timely and efficient manner adhering to critical feedback provided during code review before submission. Team leader for number bugs addressed for several releases.

2016 Electronic Arts [Contract]

Job Title: Software Engineer Intern

Position Details: Member of Madden Continuous Development team. Titles supported and contributed to include Madden NFL 17 and 18 using the Ignite and the Frostbite Game Engine. Implemented features on Blaze server.

- Implemented binder space increase feature for users subscribed to Madden subscription service. Created new RPC to fetch this data on login and when going into Ultimate Team. Implemented front end UI changes as well as back-end validation.
- Updated sets and event UI to include content driven UI to allow artist to update images on a per set/event basis.
- Implemented pause feature in new multiplayer mode to not stop the game.
- Implemented various other UI features and telemetry tracking features.

Game Engine Experience

Unreal Engine 5 and 4, Unity 5, Frostbite, Ignite

Programming/Scripting Languages

C#, Java, C++, MySQL, JavaScript, Splunk, C, Haxe.

Operating System Environment Experience

Android, IOS, Windows Mobil 6 OS, Ubuntu, Windows 10

Education

2016 University of Central Florida, MS in Interactive Entertainment

2014 University of South Florida, BS in Computer Engineering