Updated Piral Plugin Architecture Proposal

Modernized Toolchain
- **Vite 7** with **Rolldown** for bundling
- **SWC** for TypeScript/JS transpilation
- **Vitest**, **React Testing Library**, **Playwright** for testing
- **TeamCity Kotlin DSL** for CI/CD pipelines

Architecture Overview
This proposal updates the original plugin-based architecture leveraging Piral with modern build tools, testing, and deployment flows. The architecture preserves a thin app shell, independently deployable pilets, and Azure cloud infrastructure.

Key Sections
App Shell
Provides:
- Plugin discovery
- Shared services (auth, analytics)
- Routing and layout
- Error boundaries

Pilets
Self-contained micro frontends:
- Independently built and deployed
- Loaded dynamically based on user context
- Registered via API (pages, tiles, extensions)

Feed Service
Manages:
- Pilet metadata
- Versioning and rollout
- User-targeting filters

Azure Infrastructure
- Static Web Apps for shell
- Blob Storage for pilet hosting
- Front Door for global routing
- B2C for authentication

Updated Build Configuration
```ts
import { defineConfig } from 'vite'
import react from '@vitejs/plugin-react-swc'

export default defineConfig({
plugins: [react()],
build: { target:'es2022', rollupOptions:{} }
})
```

Updated Testing Stack
- **Vitest** for unit tests

- **RTL** for component tests
- **Playwright** for UI flows

Updated CI/CD with TeamCity Kotlin DSL
```kotlin
object BuildPilet: BuildType({
steps {
script { scriptContent = "npm ci" }
script { scriptContent = "npm run build" }
script { scriptContent = "npm test -- --coverage" }
}
})
```

Gliffy Diagram Definitions
Use these nodes in Confluence Gliffy:
- App Shell
- Pilet Registry (Feed Service)
- Pilet Bundles
- Azure Front Door
- Static Web App
- API Gateway
- Data Services