Contents

Φορολογικός Υπολογιστής - Project Documentation

□ Table of Contents

---

1. Project Overview

Φορολογικός Υπολογιστής (Greek Tax Calculator) is a comprehensive web application designed for Greek accountants and business owners to calculate taxes under various business scenarios. The application replicates the functionality of a complex Excel spreadsheet with 456 formulas, implementing the Greek tax system including progressive taxation, corporate tax, deemed taxation, and prepayment calculations.

Purpose

- Simplify tax calculations for Greek businesses
- Provide a database-backed solution for storing and retrieving business data
- Enable accurate tax planning across multiple income scenarios
- Replace manual Excel-based calculations with automated, validated computations

Key Business Scenarios

The application supports 7 calculation scenarios:

1. **Ατομική Επιχείρηση** (Individual Business) - Progressive tax 9%-44%
2. **Εταιρεία** (Company) - Flat corporate tax 22%
3. **Ατομική + Εταιρεία** (Combined Individual + Company)
4. **ΠΙΝΑΚΑΣ Ατομική + Εταιρεία** (Comparison Table)
5. **Ατομική & Μισθωτές** (Individual + Employee Income)
6. **Ατομική & Μισθωτές + Εταιρεία** (Full Combination)
7. **ΠΙΝΑΚΑΣ Ατομική & Μισθωτές + Εταιρεία** (Full Comparison)

---

2. Architecture

High-Level Architecture

```
                      Client Browser
  (React Components + Next.js Pages + Client State)

                      HTTPS



                   Next.js Server


     Pages           API Routes         Auth
     (SSR)                             (NextAuth)
```

```
            Calculation Engine (lib/)
     - Tax Calculations
     - Progressive Tax Brackets
     - Corporate Tax Logic
     - Deemed Taxation


                      Prisma ORM


              PostgreSQL Database
  - Users & Authentication
  - Business Information
  - Calculation Data (Individual, Company, Employee)
  - Calculation History & Export Logs
```

Design Patterns

1. **Server-Side Rendering (SSR)**: Pages fetch data on the server for better performance and SEO
2. **API Routes**: RESTful endpoints for CRUD operations
3. **Component-Based Architecture**: Reusable UI components with clear separation of concerns
4. **Type Safety**: TypeScript throughout the application
5. **Authentication Middleware**: NextAuth.js for secure session management

---

3. Technology Stack

Frontend

- **Framework**: Next.js 14.2.28 (React 18.2.0)
- **Language**: TypeScript 5.2.2
- **Styling**: Tailwind CSS 3.3.3
- **UI Components**: Custom components + Radix UI primitives
- **Icons**: Lucide React
- **State Management**: React Hooks (useState, useEffect)
- **Form Handling**: Controlled components with validation

Backend

- **Runtime**: Node.js
- **Framework**: Next.js API Routes
- **Database ORM**: Prisma 6.7.0
- **Database**: PostgreSQL
- **Authentication**: NextAuth.js 4.24.11
- **Password Hashing**: bcryptjs
- **Session Management**: JWT tokens

Development Tools

- **Package Manager**: Yarn
- **Linting**: ESLint
- **Type Checking**: TypeScript
- **Database Migrations**: Prisma Migrate

---

4. Features

□ Implemented Features

Authentication & User Management

- User registration and login
- Secure password hashing with bcryptjs
- JWT-based session management
- Protected routes and API endpoints
- Greek language error messages

Business Management

- Create new business entities
- Store business information (name, activity, address, tax ID)
- View all businesses in dashboard
- Navigate to business details
- Track last calculation date

Individual Business Calculations (Ατομική Επιχείρηση)

- **Income Statement Inputs**:
  - Gross revenues
  - Operating expenses (multiple categories)
  - Depreciation
  - Financial costs
  - Other income/expenses
- **Tax Adjustments**:
  - Non-deductible expenses
  - Tax-exempt income
  - Previous year losses
  - Other adjustments
- **Withholdings & Prepayments**:
  - Tax already withheld
  - Previous prepayments
- **Deemed Taxation (Τεκμαρτή Φορολόγηση)**:
  - Lifestyle indicators (house size, car cc, swimming pool, etc.)
  - Automatic calculation of deemed income
  - Comparison with actual income
- **Real-time Calculations**:
  - Taxable income
  - Progressive tax (9%-44% brackets)
  - Solidarity contribution (2.2%-10%)
  - Total tax due
  - Minimum card spending requirement

  - Net income
  - Prepayments for next year (55%)
- **Year-over-Year Comparison**:
  - Percentage change indicators
  - Trend arrows (up/down)
  - Previous year’s data display
- **Data Persistence**:
  - Save calculations to database
  - Load existing calculations
  - Automatic update of business last calculation date

Dashboard Features

- Display all user’s businesses
- Show last calculation year for each business
- Quick access to business details
- Create new business button
- User information display
- Logout functionality

□ Placeholder Features (To Be Implemented)

1. **Company Calculations (Εταιρεία)**
   - 22% flat corporate tax
   - Professional fees deduction
   - Dividend distribution calculations
2. **Combined Scenarios**
   - Individual + Company consolidated analysis
   - Individual + Employee income calculations
   - Full combination (all income sources)
3. **Comparison Tables (ΠΙΝΑΚΑΣ)**
   - Side-by-side scenario comparisons
   - Optimal structure recommendations
4. **Export Functionality**
   - PDF report generation
   - Excel export with formulas
   - Word document export
5. **Additional Features**
   - Multi-year historical analysis
   - Tax planning scenarios
   - Bulk data import
   - Advanced reporting

---

5. Database Schema

Entity Relationship Diagram

```
     User

  id
  name
```

```
email
password
createdAt
updatedAt

                   1:N


         Business

id
userId
businessName
activity
address
taxId
lastCalculationDate
createdAt
updatedAt

                                       1:N




Individual                 Company               EmployeeIncome
 Calculation              Calculation

id                      id                      id
businessId              businessId              businessId
year                    year                    year
grossRevenues           revenues                salary
operatingExpenses       expenses                bonuses
depreciation            professionalFees        otherIncome
... (40+ fields)        ... (30+ fields)        ... (15+ fields)



CalculationHistory           ExportLog

id                         id
userId                     userId
businessId                 businessId
calculationType            exportType
snapshot                   fileName
createdAt                  createdAt
```

Key Tables

User

- Stores user account information
- Linked to NextAuth for authentication

- One user can have multiple businesses

Business

- Core business entity
- Stores business identification and contact info
- Parent to all calculation types

IndividualCalculation

- Most comprehensive calculation model
- 40+ fields for income statement, adjustments, and results
- Supports deemed taxation calculations
- Unique constraint on (businessId, year)

CompanyCalculation

- Corporate tax calculations (22% flat rate)
- Professional fees and other deductions
- Unique constraint on (businessId, year)

EmployeeIncome

- Employee income calculations
- Can be combined with business income
- Unique constraint on (businessId, year)

CalculationHistory

- Stores snapshots of calculations (JSON)
- Audit trail for changes
- Useful for historical analysis

ExportLog

- Tracks all exports (PDF, Excel, Word)
- File metadata and timestamps

---

6. Calculation Engine

Core Functions (`lib/tax-calculator.ts`)

1. Progressive Tax Calculation

```
function calculateProgressiveTax(income: number): {
  tax: number;
  bracket: string;
}
```

Greek Tax Brackets (2024-2025):

Income Range	Rate	Tax on Range
€0 - €10,000	9%	€0 - €900
€10,001 - €20,000	22%	€900 - €3,100
€20,001 - €30,000	28%	€3,100 - €5,900
€30,001 - €40,000	36%	€5,900 - €9,500
Over €40,000	44%	€9,500+

Logic: - Applies marginal tax rates - Each euro is taxed at its bracket rate - Returns total tax and highest bracket reached

2. Solidarity Contribution

```
function calculateSolidarityContribution(income: number): number
```

Rates: - €12,000 - €20,000: 2.2% - €20,001 - €30,000: 5% - €30,001 - €40,000: 6.5% - €40,001 - €65,000: 7.5% - €65,001 - €220,000: 9% - Over €220,000: 10%

3. Corporate Tax

```
function calculateCorporateTax(income: number): number
```

Logic: Flat 22% on taxable income

4. Deemed Taxation (Τεκμαρτή Φορολόγηση)

Purpose: Prevent tax evasion by calculating minimum taxable income based on lifestyle indicators.

Indicators: - House size (€40/m² for owned, €200/m² for rented) - Car engine size (€250/cc + 15%) - Swimming pool (€3,000) - Private school tuition (€100/m² of residence) - Domestic employee (€3,000/year) - Number of dependents (reduction)

Formula:

```
DeemedIncome = Σ(lifestyle indicators) - (dependents × reduction)
TaxableIncome = MAX(actualIncome, deemedIncome)
```

5. Minimum Card Spending

```
function calculateMinimumCardSpending(income: number): number
```

Formula: MIN(income × 30%, €20,000)

Purpose: Combat tax evasion by requiring electronic payments

6. Prepayment Calculation

Rules: - Individual businesses: 55% of current year’s tax - Companies: 80% of current year’s tax - Employee income: 100% (withheld by employer)

7. Net Income Calculation

```
NetIncome = TaxableIncome
          - TotalTax
          - SolidarityContribution
          + TaxWithheld
          + Prepayments
```

Calculation Flow for Individual Business

```
1. Input Validation
   ↓
2. Income Statement Calculation
   Gross Revenues - Operating Expenses - Depreciation = Operating Result
   ↓
3. Taxable Income Before Adjustments
   Operating Result + Other Income - Financial Costs
   ↓
4. Apply Tax Adjustments
   + Non-deductible expenses
   - Tax-exempt income
   - Previous losses
   ± Other adjustments
   ↓
5. Calculate Deemed Income (Lifestyle Indicators)
   ↓
6. Determine Final Taxable Income
   MAX(Adjusted Income, Deemed Income)
   ↓
7. Calculate Progressive Tax
   Apply 9%-44% brackets
   ↓
8. Calculate Solidarity Contribution
   Apply 2.2%-10% rates
   ↓
9. Calculate Total Tax Due
   Tax + Solidarity
   ↓
10. Apply Credits
    - Tax withheld
    - Previous prepayments
   ↓
11. Calculate Net Tax Due / Refund
   ↓
12. Calculate Next Year's Prepayments
    Total Tax × 55%
   ↓
13. Calculate Net Income
   ↓
14. Calculate Minimum Card Spending
```

---

7. File Structure

```
greek_tax_calculator/
  nextjs_space/
      app/                           # Next.js App Router
          api/                      # API Routes
              auth/
                  [...nextauth]/
                      route.ts      # NextAuth API handler
```

```
        signup/
            route.ts          # User registration endpoint
        business/
            route.ts          # Business CRUD operations
        calculation/
            individual/
                route.ts      # Individual calculation save/fetch
    auth/                     # Authentication Pages
        login/
            page.tsx          # Login page
        signup/
            page.tsx          # Registration page
    dashboard/
        page.tsx              # Main dashboard
    business/
        new/
            page.tsx          # Create new business
        [id]/
            page.tsx          # Business detail & scenarios
    calculation/
        [businessId]/
            individual/
                page.tsx      # Individual calculation (IMPLEMENTED)
            company/
                page.tsx      # Company calculation (PLACEHOLDER)
            combined/
                page.tsx      # Combined (PLACEHOLDER)
            employee/
                page.tsx      # Employee income (PLACEHOLDER)
            full/
                page.tsx      # Full combination (PLACEHOLDER)
    layout.tsx                # Root layout
    page.tsx                  # Landing page
    globals.css               # Global styles
components/                   # React Components
    auth/
        login-form.tsx
        signup-form.tsx
    business/
        business-info.tsx
        calculation-scenarios.tsx
        new-business-form.tsx
    calculation/
        individual-calculation-form.tsx
    dashboard/
        dashboard-header.tsx
        business-list.tsx
    ui/                       # Reusable UI components
        button.tsx
        input.tsx
        label.tsx
        card.tsx
        ... (40+ components)
    providers.tsx
    theme-provider.tsx
lib/                          # Utility Libraries
    auth-options.ts           # NextAuth configuration
    db.ts                     # Prisma client
    tax-calculator.ts         #   CORE CALCULATION ENGINE
    types.ts                  # TypeScript interfaces
    utils.ts                  # Helper functions
prisma/
    schema.prisma             # Database schema
scripts/
    seed.ts                   # Database seeding
types/
    next-auth.d.ts            # NextAuth type extensions
public/                       # Static assets
    favicon.svg
    og-image.png
    robots.txt
.env                          # Environment variables (NOT in git)
package.json                  # Dependencies
tsconfig.json                 # TypeScript config
tailwind.config.ts            # Tailwind CSS config
next.config.js                # Next.js config
```

---

8. API Endpoints

Authentication

`POST /api/auth/callback/credentials` **NextAuth endpoint for login** - Body: `{ email, password }` - Returns: JWT token in cookie - Errors: 401 Unauthorized

`POST /api/signup` **User registration** - Body: `{ name, email, password }` - Returns: `{ id, name, email }` - Errors: 400 (validation), 409 (user exists)

Business Management

`POST /api/business` **Create new business** - Auth: Required - Body: `{ businessName, activity, address, taxId }` - Returns: Created business object - Validation: 9-digit tax ID

`GET /api/business` **Get all user's businesses** - Auth: Required - Returns: Array of businesses with latest calculations

Calculations

`POST /api/calculation/individual` **Save individual calculation** - Auth: Required - Body: Full calculation input object (40+ fields) - Returns: Saved calculation with ID - Logic: Upserts based on (businessId, year)

`GET /api/calculation/individual?businessId=X&year=Y` **Fetch individual calculations** - Auth: Required - Query: businessId (required), year (optional) - Returns: Array of calculations

---

9. Authentication Flow

Registration Flow

```
User → SignupForm → POST /api/signup
                        ↓
                  Validate input
                        ↓
                  Hash password (bcryptjs)
                        ↓
                  Create user in DB
                        ↓
                  Auto-login (signIn)
                        ↓
                  Redirect to /dashboard
```

Login Flow

```
User → LoginForm → signIn('credentials', {...})
                        ↓
                  NextAuth validates
                        ↓
                  Compare password hash
                        ↓
                  Create JWT session
                        ↓
                  Set HTTP-only cookie
                        ↓
                  Redirect to /dashboard
```

Protected Route Pattern

```
// In page.tsx
const session = await getServerSession(authOptions);
if (!session) {
  redirect('/auth/login');
}

// In API route
const session = await getServerSession(authOptions);
if (!session?.user?.id) {
  return NextResponse.json(
    { error: 'Unauthorized' },
    { status: 401 }
  );
}
```

---

10. Development Guidelines

Code Style

- **TypeScript**: All files must be .ts or .tsx
- **Naming**: camelCase for variables, PascalCase for components

- **Components**: Functional components with hooks
- **Async Operations**: Use async/await, not .then()
- **Error Handling**: Try-catch in API routes, error states in components

Adding New Calculation Scenarios

1. **Update Prisma Schema**: Add new model if needed
2. **Create Calculation Function**: In `lib/tax-calculator.ts`
3. **Add API Route**: `app/api/calculation/[scenario]/route.ts`
4. **Create Form Component**: `components/calculation/[scenario]-form.tsx`
5. **Create Page**: `app/calculation/[businessId]/[scenario]/page.tsx`
6. **Update Types**: Add interfaces to `lib/types.ts`
7. **Test**: Verify calculations against Excel

Database Migrations

```
# After schema changes
cd nextjs_space
yarn prisma generate
yarn prisma db push

# For production
yarn prisma migrate dev --name describe_your_change
```

Testing Checklist

☐ User can register and login
☐ Protected routes redirect unauthenticated users
☐ Business CRUD operations work
☐ Calculations match Excel results
☐ Data persists correctly
☐ Year-over-year comparisons accurate
☐ Form validation prevents invalid data
☐ Error messages display in Greek
☐ Responsive design works on desktop

Performance Considerations

- Server-side data fetching reduces client-side requests
- Calculations run in memory (no database queries)
- Database indexes on userId, businessId, year
- JWT sessions avoid database lookups on every request

Security Best Practices

- ▯ Passwords hashed with bcryptjs
- ▯ JWT tokens HTTP-only cookies
- ▯ API routes check authentication
- ▯ Business ownership verified before access
- ▯ SQL injection prevented by Prisma
- ▯ Input validation on client and server
- ⚠ TODO: Rate limiting on API routes
- ⚠ TODO: CSRF protection

- ⚠ TODO: XSS sanitization for user inputs

Roadmap

Phase 1 (□ Complete)

- Authentication system
- Business management
- Individual calculation with deemed taxation
- Database persistence
- Year-over-year comparison

Phase 2 (□ In Progress)

- Company calculations
- Combined scenarios
- Comparison tables

Phase 3 (□ Planned)

- Export to PDF, Excel, Word
- Advanced reporting
- Multi-year analysis
- Tax planning simulator

Phase 4 (□ Ideas)

- Multi-user support (accounting firms)
- Client portal
- Automated tax filing integration
- Mobile app
- Real-time tax law updates

Support

For technical questions or issues: - Review this documentation - Check the Excel Technical Specification - Consult the User Manual - Review source code comments

---

Document Version: 1.0
Last Updated: December 19, 2025
Author: Cnstech.gr