

Презентація кваліфікаційної
роботи
на тему:

Розробка веб-застосунку з
використанням фреймворку
React

студента III курсу

Спеціальності Комп'ютерні Науки

Грачова Олександра

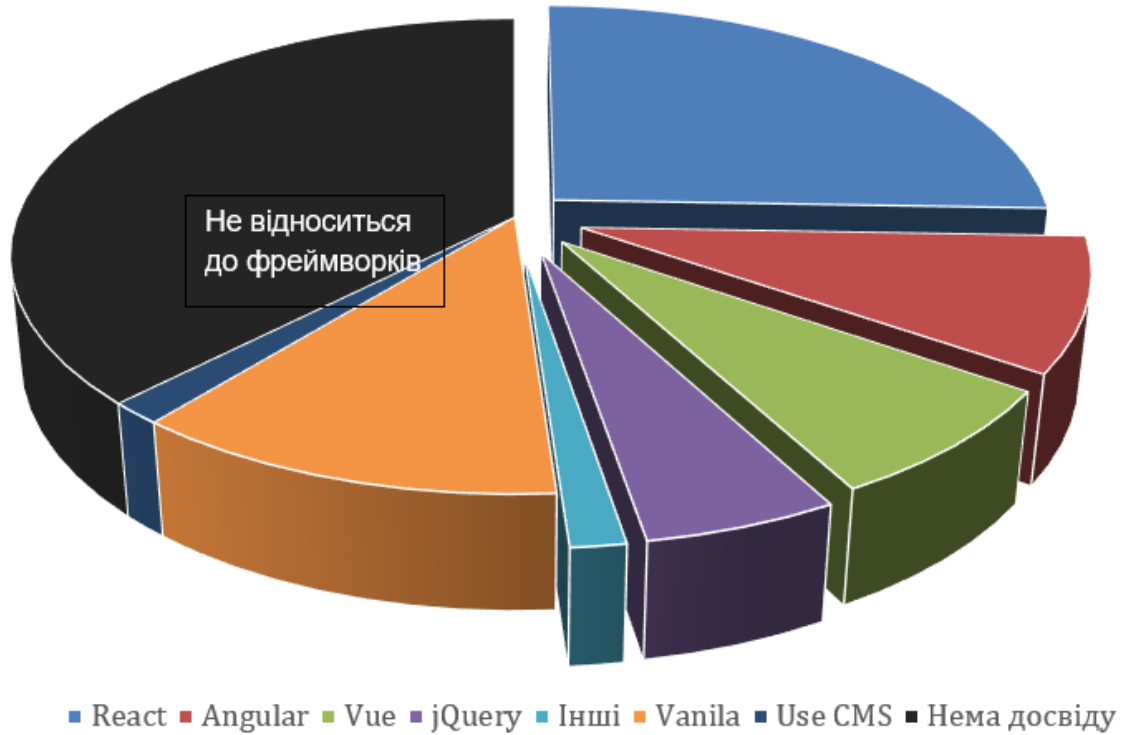
Історія

- React – один із найпопулярніших JS фреймворків
- Перший серйозний фреймворк
- Заснував революційно новий підхід з віртуальним DOM

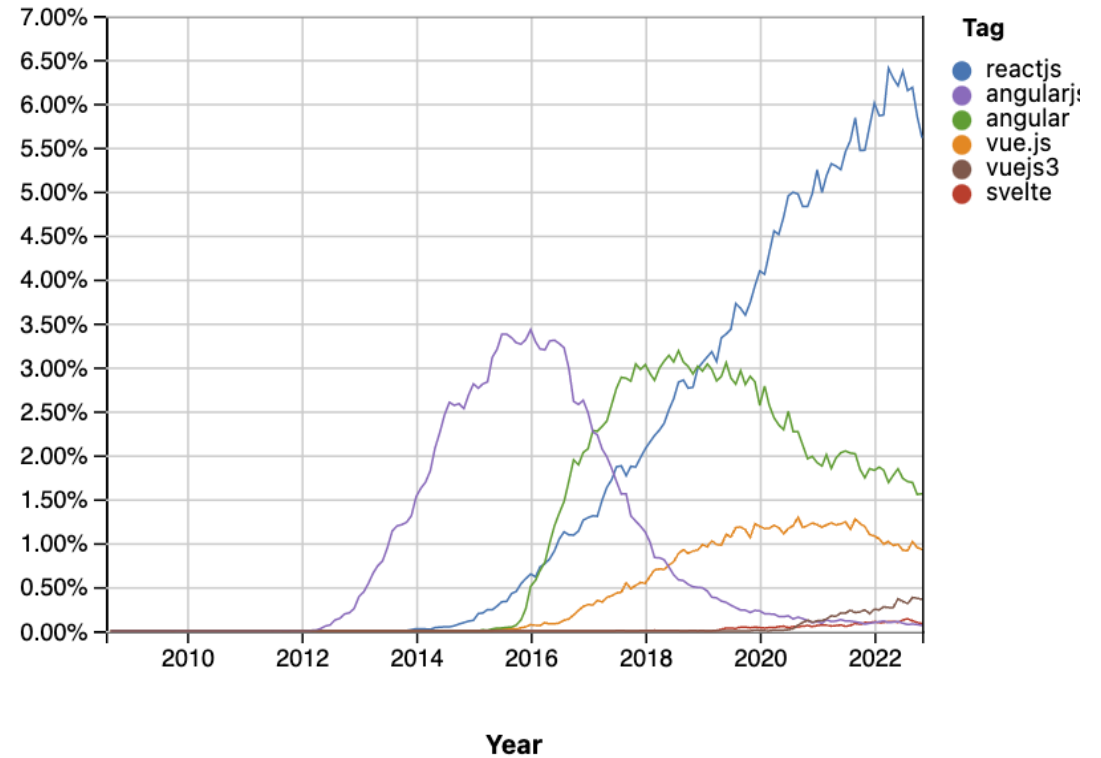


Предметна область

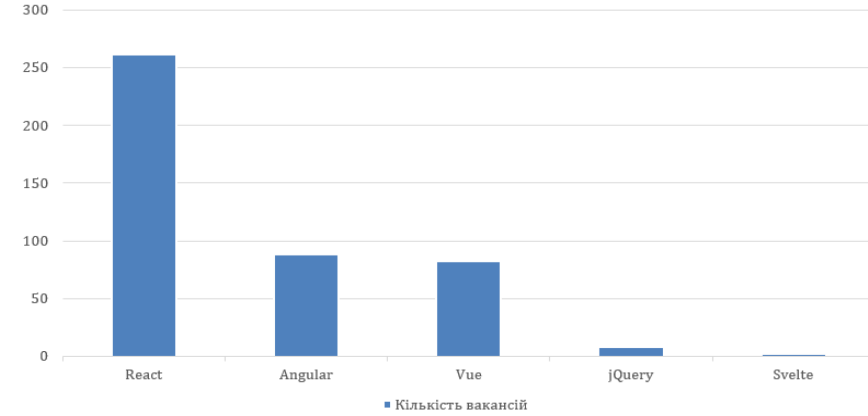
Якими інструментами ви найбільше користуєтесь



% of Stack Overflow questions that month



Ринок праці
Кількість вакансій



Переваги та недоліки

- Ефективність
- Компонентна архітектура
- Масштабованість
- Розширюваність
- Підтримка
- Навчання
- Несумісність зі старим кодом
- Великий розмір

Основні технічні особливості

- Компоненти (класові\функційні)
- JSX
- Стани компонентів
- Хуки (hooks)
- Аjax
- НОС

Створення застосунку

Username:

Password:

Login



Next player: X

1. To start of game

Coords: 1, 1: X

2. Go to move #1

Coords: 1, 2: O

3. Go to move #2

Coords: 2, 2: X

4. Go to move #3

Coords: 2, 3: O

5. Go to move #4

Logout

Сторінка входу

```
return (  
  <div className='main-block-auth'>  
    <form className='form-auth' ref={refForm}>  
      <div>  
        <label className='label-email' htmlFor="email">Username:</label>  
        <input className="input-email" type="text" value={email} onChange={handleUsernameChange}/>  
      </div>  
      <div>  
        <label className='label-email' htmlFor="password">Password:</label>  
        <input className="input-password" type="password" value={password} onChange={handlePasswordChange}/>  
      </div>  
      <button className='button-login' type="button" onClick={handleOnLogin}>Login</button>  
      {error && <p>{error}</p>}  
    </form>  
  </div>  
>);
```

```
const handleOnLogin = (event: MouseEvent<HTMLButtonElement>) => {  
  axios.post(`${api_url}/auth`, { email, password })  
    .then(response => {  
      if (response.status === 200) {  
        localStorage.setItem("userData", response.data.token);  
        navigate('/');  
      }  
    })  
    .catch(error => {  
      setError('Invalid username or password');  
    });  
};
```

Захист роутів

```
export function ProtectedRoute() {  
  const token = localStorage.getItem('userData');  
  
  return token ? <Outlet /> : <Navigate to="/login" />;  
};
```

```
export function UseRoutes (props: any) {  
  return(  
    <Routes>  
      <Route path="/" element={<HomePage />}/>  
  
      <Route path="/game" element={<ProtectedRoute />}>  
        <Route path="/game" element={<TicTacToePage />}/>  
      </Route>  
  
      <Route path="/login" element={<LoginPage />}/>  
      <Route path="*" element={<Page404 />}/>  
    </Routes>  
  )  
}
```

Крестики нулики

```
return (<div className="game">  
  <div className="game-board">  
    <Board squares={currentState.squares} onClick={({i: number}) => handleClick(i)} />  
  </div>  
  <div className="game-info">  
    <div>{status}</div>  
    <ol>{moves}</ol>  
  </div>  
  <button className='button-logout' onClick={handleLogout}>Logout</button>  
</div>  
);
```

```
const moves = history.map((step, move) => {  
  const desc = move ? `Go to move #${move}` : 'To start of game';  
  return (<li key={move}>  
    <button className='button-history' onClick={() => jumpTo(move)}>{desc}</button>  
    <h4>Coords: {coordHistory[move]}</h4>  
  </li>  
);  
});
```

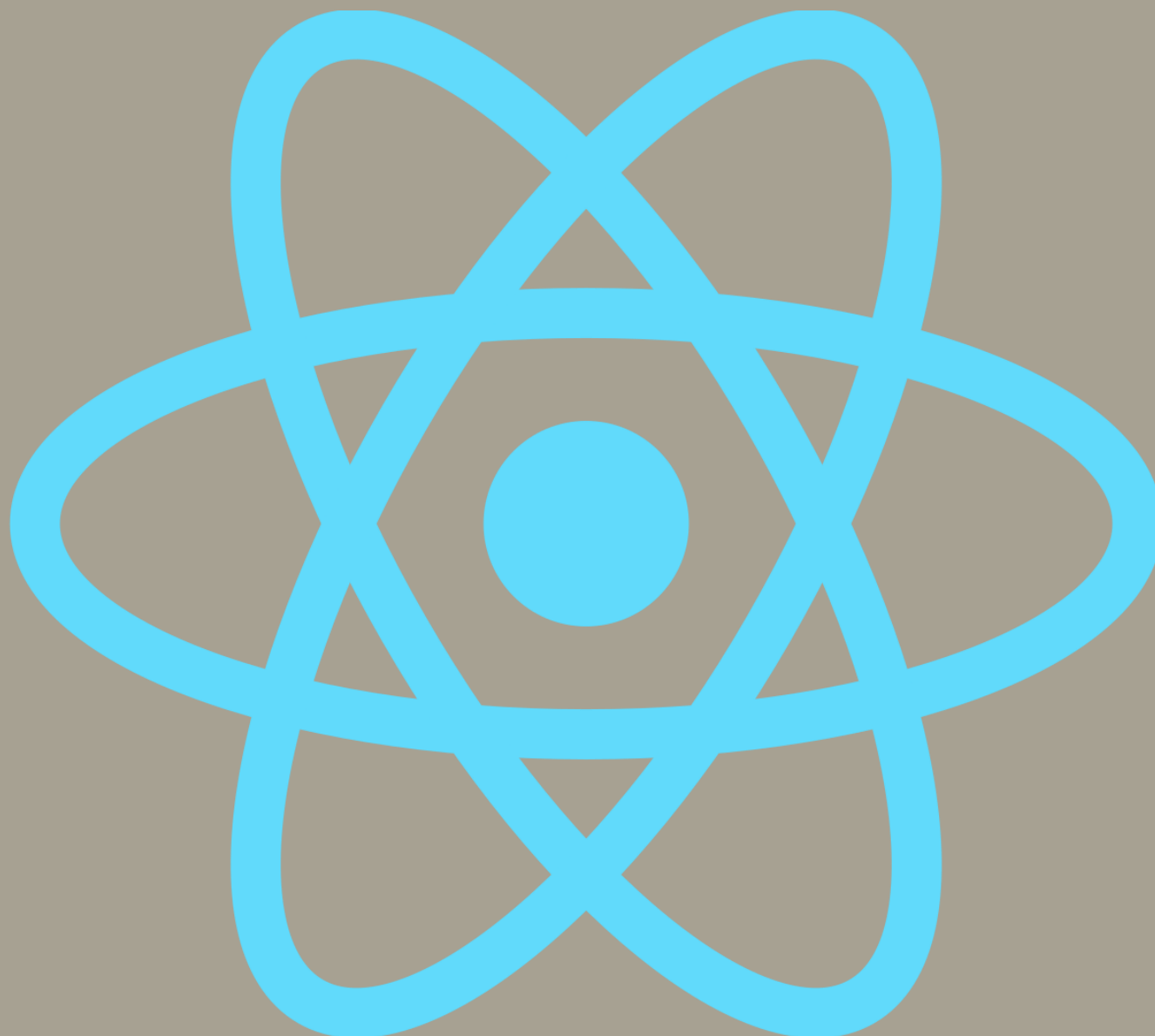
```
const handleClick = (i: number) => {  
  const currentHistory = history.slice(0, stepNumber + 1);  
  const currentState = currentHistory[currentHistory.length - 1];  
  const squares = [...currentState.squares];  
  
  const currentCoordHistory = coordHistory.slice(0, stepNumber + 1);  
  if (squares[i] || calculateWinner(squares)) return;  
  
  squares[i] = xIsNext ? CROSS : CIRCLE;  
  setHistory([...currentHistory, { squares }]);  
  setCoordHistory([  
    ...currentCoordHistory,  
    `${getRow(i)}, ${getCol(i)}: ${squares[i]}`,  
  ]);  
  setStepNumber(currentHistory.length);  
  setXIsNext(!xIsNext);  
};
```

Компонент доски

```
interface BoardProps {  
  squares: CellValue[];  
  onClick: (i: number) => void;  
}
```

```
export function Cell(props: any) {  
  return (  
    <button className='square' onClick={props.onClick}>  
      {props.value}  
    </button>  
  )  
}
```

```
export function Board(props: BoardProps) {  
  const renderCell = (i: number) => {  
    return (  
      <Cell  
        value={props.squares[i]}  
        onClick={() => props.onClick(i)}  
      />  
    );  
  };  
  
  return (  
    <div>  
      <div className="board-row">  
        {renderCell(0)}  
        {renderCell(1)}  
        {renderCell(2)}  
      </div>  
      <div className="board-row">  
        {renderCell(3)}  
        {renderCell(4)}  
        {renderCell(5)}  
      </div>  
      <div className="board-row">  
        {renderCell(6)}  
        {renderCell(7)}  
        {renderCell(8)}  
      </div>  
    </div>  
  );  
}
```



Висновок

Дякую за увагу

Джерела презентації

- Іконка реакту - <https://commons.wikimedia.org/wiki/File:React-icon.svg?uselang=ua>