Our Team

Tamanna Hussain
Project Manager

Kamrul Hossain
Systems Architect

Ramisa Murshed
Language Guru

Xabi Peralta
Tester
Table of Contents

01 Introduction
Motivation, intended audience, use cases, etc.

02 Features
Data types, function / variable declarations, etc.

03 Implementation
Software technologies, architecture

04 Future Work
Additional features to implement

05 Demonstration
Demo of sample program
Introduction

QWEB is a website language inspired by interactive visualization languages such as Processing.js. Its purpose is to help both novel and experienced programmers design and develop websites to run in their browser.
Motivation

To create an alternative website language that is more intuitive, maximizes human readability, and incorporates familiar programming constructs that are used in traditional high-level programming languages.
01 Language Features
Brief overview of data types, function and variable declarations, etc.
## Language Features

<table>
<thead>
<tr>
<th>Data Types</th>
<th>Variable Declaration</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td><code>int a;</code></td>
</tr>
<tr>
<td>Integer</td>
<td><code>a = 50;</code></td>
</tr>
<tr>
<td>Float</td>
<td><code>bool b;</code></td>
</tr>
<tr>
<td>Boolean</td>
<td><code>str c;</code></td>
</tr>
<tr>
<td>Void</td>
<td><code>c = &quot;Hello World&quot;</code></td>
</tr>
</tbody>
</table>
Language Features

Function Definition

```plaintext
function int check(bool b)
{
    int a;
    a = 3;
    if (b)
        print(a)
    output a;
}
```

Function Call

```plaintext
function int main(){
    print(check(true));
    print(check(false));
}
```
Language Features

Built-in Functions

- createHeader()
- createParagraph()
- createSubheader()
- createList()
- createImage()

*modifications to print functions
Implementation

Software technologies, architecture
Architecture

Scanner

Parser

Abstract Syntax Tree

Semantic Checker

Code Generation

LLVM
03 Future Work

Additional features to implement
Future Work

01 Data Types (Lists, Shapes, etc.)
02 Object-Oriented Programming
03 CSS Features
04 Pseudocode-Style Syntax
**Lessons Learned**

**Project Idea**
Brainstorm early on and have a robust idea of plausible features to implement.

**Accountability**
Make a project schedule with internal group deadlines and hold each other accountable.

**Office Hours**
Attend office hours in addition to meetings with assigned TA.

**Milestones**
Meet each of the class deadlines for project milestones; fix bugs sooner than later.
Demo
THANK YOU!